DEVELOPMENT OF A SMARTGRID PROTOTYPE FOR THE PROPOSED 33 KV DISTRIBUTION SYSTEM IN NIT ROURKELA

A THESIS SUBMITTED IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE DEGREE OF

Bachelor of Technology in

ELECTRICAL ENGNEERING

By

Aurabind Pal 107EE004

Anubhav Ratha 107EE032

Vaibhav Mishra 107EE034

Anshul Garg 107EE061

Department of Electrical Engineering
National Institute of Technology
Rourkela

2011

DEVELOPMENT OF A SMARTGRID PROTOTYPE FOR THE PROPOSED 33 KV DISTRIBUTION SYSTEM IN NIT ROURKELA

A THESIS SUBMITTED IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE DEGREE OF

Bachelor of Technology in

ELECTRICAL ENGNEERING

By

Aurabind Pal 107EE004

Anubhav Rath 107EE032

Vaibhav Mishra 107EE034

Anshul Garg 107EE061

Under the Guidance of

Prof. Susmita Das

Department of Electrical Engineering
National Institute of Technology, Rourkela

DEVELOPMENT OF A SMARTGRID PROTOTYPE FOR THE PROPOSED 33 KV DISTRIBUTION SYSTEM IN NIT ROURKELA

National Institute of Technology, Rourkela

CERTIFICATE

This is to certify that the thesis entitled **"Development of a smart grid for the proposed 33 KV ring main Distribution System in NIT Rourkela"** submitted by **Aurabind Pal (107EE004), Anubhav Ratha (107EE032), Vaibhav Mishra (107EE034), Anshul Garg (107EE061)** in the partial fulfillment of the requirement for the degree of **Bachelor of Technology** in **Electrical Engineering**, National Institute of Technology, Rourkela, is an authentic work carried out by them under my supervision.

To the best of my knowledge the matter embodied in the thesis has not been submitted to any other university/institute for the award of any degree or diploma.

Date:

(Prof. Susmita Das)
Dept of Electrical Engineering
National Institute of Technology
Rourkela-769008

DEVELOPMENT OF A SMARTGRID PROTOTYPE FOR THE PROPOSED 33 KV DISTRIBUTION SYSTEM IN NIT ROURKELA

ACKNOWLEDGEMENT

We wish to express our profound sense of deepest gratitude to our guide and motivator *Prof. Susmita Das*, Electrical Engineering Department, National Institute of Technology, Rourkela for her valuable guidance, sympathy and co-operation and finally help for providing necessary facilities and sources during the entire period of this project.

We wish to convey our sincere gratitude to *Prof. Y.K. Sahu* of Electrical Engineering Department, who provided us with all the documents regarding the 33KV project and arranged a visit to Rourkela Steel Plant where the SCADA system is already implemented. The facilities and co-operation received from the technical staff of Electrical Engineering Department is thankfully acknowledged.

Last, but not least, we would like to thank the authors of various research articles and book that we referred to during the course of the project.

Aurabind Pal
Anubhav Ratha
Vaibhav Mishra
Anshul Garg

CONTENTS

ABSTRACT

The non-reliability of fossil fuels has forced the world to use energy efficiently. These days, it is being stressed to use the electrical power smartly so that energy does not go waste. And hence comes the concept of a Smart Grid. So it becomes necessary for reputed places of academics to develop the prototype of the same in their campus.

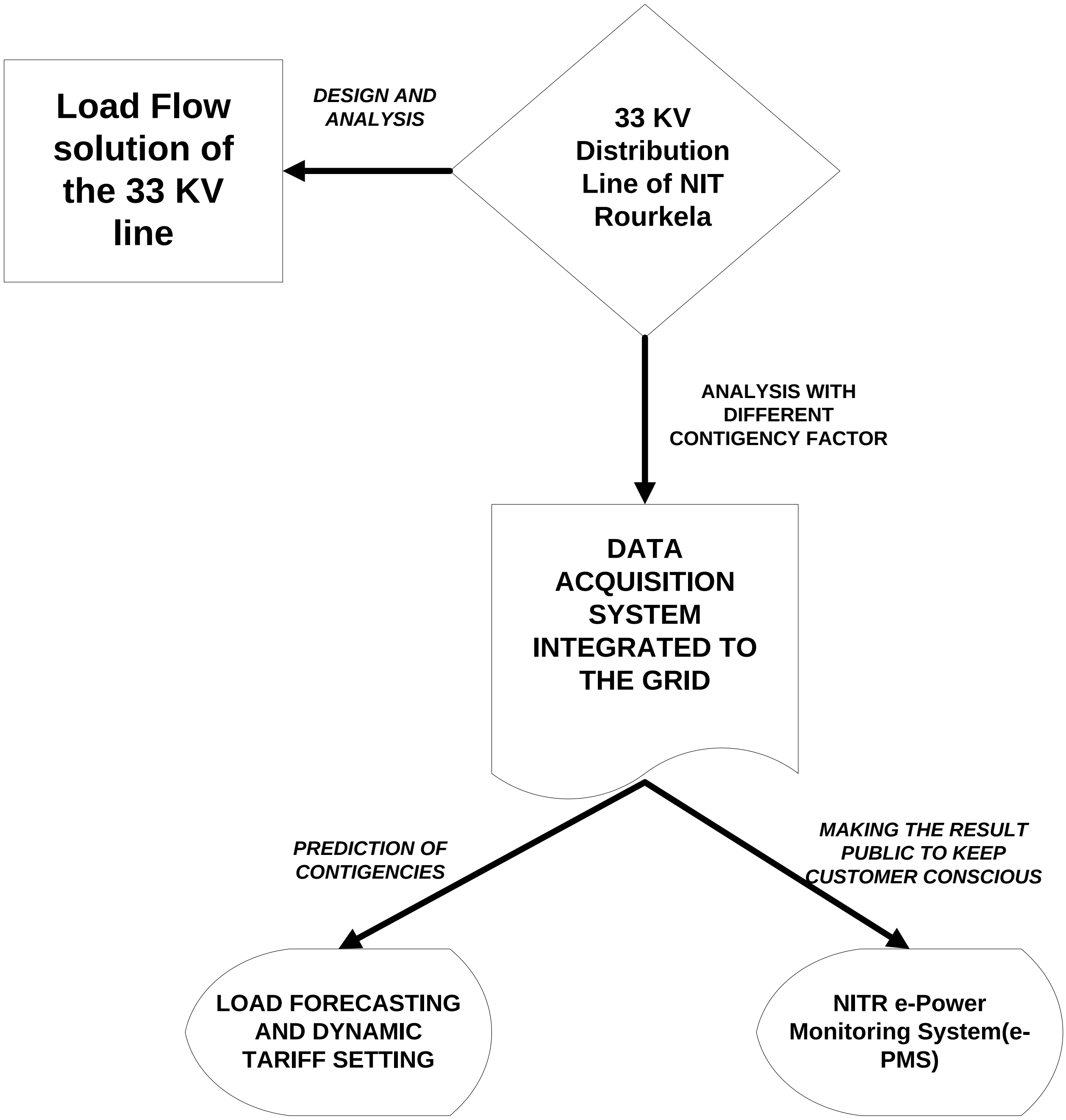
National Institute of Technology (NIT) Rourkela intends to set up a 33KV Ring Main Distribution System including 33/0.433 KV substations in its campus. The present 11KV line will be discarded and replaced by the 33KV system. The main driving force behind this step by the management is to accommodate the stupendously increased power requirement of the institute. The above mentioned plan also includes, set up of *Data Acquisition System (DAS)* that intends to monitor the electrical equipment in the substations. This is being done not only to increase the accountability and reliability of the distribution system but also to encourage academic research in the distribution automation domain. All in all, an excellent step towards make the *Grid, Smart.*

In this project work the focus is laid on getting load flow solution of the 33KV ring main system. Here the authors use a specialized algorithm for distribution network with high R/X value to obtain the load flow solution. Then using artificial neural networks computation, algorithms are implemented to do the load forecasting and dynamic tariff setting. At the end a Web Portal, the NITR e-Power Monitoring System is developed that will be an excellent interface to the public in general and will help the students of the institute to know their grid well. In short a conscious effort is put to make the grid more interactive.

Problem Statement and Project Flow Diagram

Project Goals:

- ✓ To find the Load Flow Solution of the 33 KV Ring main system.
- ✓ To declare next day power tariff rates to customers based upon ANN based Load Forecasting.
- ✓ To develop and locally host a NITR Power Monitoring Website.



LIST OF FIGURES

LIST OF TABLES

Chapter 1

The 33-KV Ring Main System of NIT Rourkela: An Introduction

1.1 Motivation

National Institute of Technology (NIT) intends to set up a 33KV Ring Main Distribution System including 33/0.433 KV substations in its campus. The present 11KV line will be discarded and replaced by the 33KV system. The main driving force behind this step by the management is to accommodate the stupendously increased power requirement of the institute. The above mentioned plan also includes, set up of *Data Acquisition System (DAS)* that intends to monitor the electrical equipment in the substations. This is being done not only to increase the accountability and reliability of the distribution system but also to encourage academic research in the distribution automation domain. All in all, an excellent step towards make the *Grid, Smart.*

The main objective of DAS is to collect the data (Voltage, Current, Active Power, Reactive Power and Frequency, Phase) from the substations and store it in the Central Master Control Server. The data stored in the server that shall be interfaced with the existing server of NIT Rourkela. The available data can be used for various kinds of analysis and decision making, using available *Artificial Intelligence* methods (Expert Systems, Artificial Neural Network, Fuzzy Inference Systems, and Genetic Algorithms). This will make the distribution system more reliable, robust and accountable.

In this project work, the task of load flow analysis of the ring main system, calculation of line losses in different contingencies, load forecasting and dynamic tariff setting using artificial neural network has been accomplished.

The objective of this project has been to develop a prototype of a smart distribution utility: a utility that is more accountable, more reliable, and more responsible and keeps its customer more aware of their consumption and ways to conserve power. This project shall be instrumental in crafting the way for automation in Indian grids, results of the studies conducted at this Institute level small scale system can be extrapolated for use in the whole Power Grid.

1.2 General Technical Specification Of The 33KV Ring Main System

Due to the enormous increase in electrical load with increasing civil infrastructure, a visionary decision was made to discard the present 11KV line with a 33KV line that will distribute power to the NIT campus. 33KV power will be received through a single feeder from WESCO at 33KV *Main Receiving Substation* (MRSS). 33KV ring main formation will be made through 33KV over head line as well as by underground cables to feed 9 nos. 33/0.433 KV substations *Loop In-Loop Out*. [1]

The layout of the substations are as follows:

- *Power tapping from WESCO Substation:* Power at 33KV will be tapped from existing WESCO substation shall be extended to the MRSS by cable.
- *33 KV MRSS:* New substation comprising of 3 nos. 33 KV outdoor air insulated bays, control room, station service transformer and boundary walls.

- *Substation-1:* 1x 500KVA, 33/0.433KV to be built within existing boundary wall of 11KV substation-1. Existing 415 V DB and DB room shall be reutilized. Substation will feed the load of colony.
- *Substation-2:* 1x500KVA, 33/0.433KV substation to be built within the existing boundary wall of 11 KV Pump house substation. New DB room shall be constructed. Load of colony and pump house will be supplied.
- *Substation-4:* 1x500KVA, 33/0.433KV to be built within existing boundary wall of 11KV substation 4. Existing 415V DB and DB room shall be extended. The substation will feed the loads of HV lab and Hall 6 extension.
- *Substation-5:* New substation comprising of 2x750KV, 33/0.433KV transformers shall be constructed along with DB rooms, cable trench, boundary wall etc. The substation will be adjacent to Computer Science Department. The substation will feed the existing load of CS Department and new loads of the Electrical Sciences buildings.
- *Substation-6&9:* Combined substation of 1x500KVA & 2x750KVA, 33/0.433KV to be built in the area adjacent to existing 11KV Substation-6 and the Dhirubhai Ambani Hall. Existing 415V DB and DB room shall be extended. The substation will be feed the loads of Dhirubhai Ambani Hall extension.
- *Substation-7:* 1x500KVA, 33/0.433KV substation to be built adjacent to existing 11KV substation-7. Existing boundary wall shall be extended to accommodate the 33KV substation. New DB room shall be constructed. The substation will feed the loads of colony and D flats.
- *Substation-8:* New substation comprising of 2x750KVA, 33/0.433KV transformers shall be constructed along with DB rooms, cable trench, boundary walls etc. The substation shall be built adjacent to new Bio-Medical building and shall feed the

loads of the BM/BT Departments, Lecture Complex, Mechanical Engineering and the Golden Jubilee building.

- *Substation-10:* New substation comprising of 2x750KVA, 33/0.433 transformers, indoor 33KV switchgears, 415V DB shall be installed within the chiller plant building. This substation shall feed the loads of chiller plant and the auxiliaries.

Having understood the configuration of the planned 33KV *Ring Main System* we are in a position to venture into designing aspects. And *Load Flow Analysis* heads the group.

Chapter 2

Load Flow Analysis of the 33-KV Ring Main System

In order to achieve the target of creating a smart distribution system, strategic planning for the network needs to be employed. The first step towards achieving any reliable system is proper planning keeping the specific goals in mind.

2.1. Planning Distribution Networks

The planning and design of electricity distribution networks can be divided into three areas:

a) *Strategic or Long Term Planning*: Deals with future major investments and main network configurations.
b) *Network Planning or Design*: Covers individual investment in the near future.
c) *Construction Design*: Structural design of each network component taking account of the various materials available.

Good system planning and design requires a sound knowledge of the existing electrical system to provide a firm base on which to assess projects for future network development. One such inevitable tool is load flow analysis. For distribution networks AC load flow studies are necessary to determine the capability of a network in all loading conditions and network configurations. This includes taking account of the loss of one or more circuits or items of equipment including the in-feed power sources, whether from generation within the network or from transformation substations where the in-feed power is obtained from a higher –voltage network. Present MV and LV network is operated in ring.

The power flow through each section of a network is influenced by the disposition and loading of each load point, and by the system losses. Maximum demand indicators installed at MV network in-feeds provides the minimum amount of load data required for system analysis. More detailed loading information is obtained in real time basis using DAS. In order to carry out power flow studies on MV and LV networks to apply correction factors to individual loads. This is because summating the maximum value of all the loads will result in too high a value for the total current flows, and therefore the overall voltage drop, if the loads do not peak at the same time. It is therefore necessary to de-rate each individual load so that the summation of the individual loads equals the simultaneous maximum demand of the group of loads. This is achieved by applying a *coincidence factor,* which is defined as the ratio of the simultaneous maximum demand of a group of load points to the sum of the maximum demands of the individual loads. The inverse of the coincidence factor is termed *diversity factor*. If kWh consumption information is available then empirical formulas or load curve synthesis can be used to determine demands at network node points.

The operation and planning studies of a distribution system demands a steady state condition of the system for various load demands and different contingencies factor. The steady-state operating condition of a system can be obtained from the load flow solution of the distribution network. If some of the variables representing the state in the load flow solution exceed their limits, certain corrective actions such as static compensators or capacitor banks, transformer tap settings etc. must be taken to stir the state variables within an acceptable and secured operating zone. For some severe violations, the corrective actions may not be adequate and certain drastic action such as load shedding must be accomplished. For a secured system, sometimes, it may be necessary to reconfigure the system to reduce the losses. The above process requires several load flow solutions with various network configurations, control variables and load demands. The efficiency of the entire process depends heavily on the efficiency and capability of the load flow program used for this purpose. Load flow analysis is a chief function of *Energy Management System and Distribution Management system.*

There are several efficient algorithms that have been developed for load flow analysis of transmission network of a high magnitude of voltage. However, these algorithms may not maintain their efficiency and reliability when applied to a low voltage distribution network. Only a few algorithms have been developed for the load flow solution of a distribution network. In general, a distribution system is fed at one point and the branches of the system have a wide range of R and X values. Also the R/X ratios of branches in a distribution system are relatively high compared to a transmission system. This makes a distribution system ill-conditioned. That is why the conventional *Newton- Raphson (NR)* method, the *Fast Decoupled Load Flow (FDLF)* method and their modifications are not suitable for solving the load flow problem of such an ill-conditioned system. For most of the cases, the NR and FDLF methods failed to converge in solving the load flow problem of distribution systems . These algorithms are not suitable for a mesh network (which has some loops). Mesh networks are not uncommon in distribution systems. However, a loop in a mesh network can be opened by adding a dummy or fictitious bus. The breaking point of a loop is called the loop break point (LBP) . The power flow through the branch that makes a loop can be simulated by injecting the same power at the LBPs. By adding some dummy buses, it is possible to convert a mesh network into a radial network. In this case, the number of dummy buses should be the same as the number of loops in the original mesh network. Thus the load flow problem of a mesh network can be solved by using the techniques of a radial network, but a proper calculation of power injections at the LBPs is required.

2.2 Generalized View of A Distribution Network

In general, a distribution system is fed at only one point and the configuration of the system is usually radial. For a radial distribution system, the number of branches (n_{brn}) and the number of buses (n) are related through a mesh network is not uncommon in a distribution system. Sometimes a mesh configuration is used to increase the efficiency, balance the load and maintain a proper voltage profile in the system. It is also used to improve the supply reliability. For a mesh network n 5 nb. The number of loops nLp of a mesh network is given by

$n=n_{brn}+l$

$n_{lup} = n_{brn} - n + 1$

A mesh network having n_{lup} loops can be reconfigured to an equivalent radial network by adding n_{lup} dummy buses shows a network in which the branch between buses f and g makes a loop. The loop of the network can be opened by adding a dummy bus g' as shown in Fig. 2. The behavior or characteristics of the original network can be preserved by injecting complex power at buses g and g' in the equivalent radial network . Note that the power injections at the LBPs (buses g and g') are equal but opposite in sign.

Fig 2.1: The breaking of the loop and creation of a dummy loop.

Network Layout

To derive the proposed load flow algorithm in a systematic way, it is required to number the branches and order the buses of the network in a particular fashion. The procedure of numbering the branches and ordering the buses is described in the following sections. Note that the dummy buses added in the mesh-radial conversion process are identified by adding a prime (') sign. For example, g' is a dummy bus added in the equivalent radial network.

Branch Numbering

The branch numbering process of a network requires the construction of a tree of the network. The tree is constructed in several layers and it starts at the root bus where the source is connected . The root bus is the swing or slack bus of the network. The first layer consists of all branches that are connected to the root bus. The next (second) layer consists of all branches that are connected to the receiving end bus of the branches in the previous (first) layer and so on. All branches of the network should be considered in the tree and they should appear only once. During the tree construction process, if it is found that the receiving end bus of a newly added branch has already been considered in the tree, it should be numbered by adding a prime sign. This implies that the newly added branch makes a loop in the network and it is opened by adding a dummy bus. The branch numbering process starts at the first layer. The numbering of branches in any layer starts only after numbering all the branches in the previous layer.

Load Flow Equations

The load flow problem of a single source network can be solved iteratively from two sets of recursive equations. The recursive equations in backward and forward directions are derived as follows. Consider that the branch *i* in a tree is connected between buses k and m. Bus k is closer to the root bus. The series impedance and shunt admittance of the branch are (R_i+jX_i)and y_i respectively. The π-circuit model of the branch is shown . The active (P) and reactive power flow through the series impedance of the branch can be written as:

$$\mathbf{P}_i' = \mathbf{P}_m^L + \mathbf{P}_m^F - \mathbf{P}_m^I$$

$$\mathbf{Q}_i' = \mathbf{Q}_m^L + \mathbf{Q}_m^F - \mathbf{Q}_m^I - \frac{\mathrm{V}_m^2 y_i}{2}$$

Here, the superscripts L, F and I in P and Q represent the load, flow and injection, respectively. The flow $P_i(Q_l)$ is the sum of the active (reactive) power flow through all the downstream branches that are connected to bus m. The procedure of finding the power injections (P^L and Q^L) at the LBPs is described in the next Section. The active (P_i) and reactive (Q_i) power flow through branch i near bus k can be written as:

$$P_i = P_i^{'} + R_i \frac{P_i^{'2} + Q_i^{'2}}{V_m^2}$$

$$Q_i = Q_i^{'} + R_i \frac{P_i^{'2} + Q_i^{'2}}{V_m^2} - \frac{V_k^2 y_i}{2} \tag{1}$$

Then the voltage magnitude can be written as:

$$V_m = \sqrt{V_k^2 - 2(P_i^{"}R_i + Q_i^{"}X_i) + (P_i^{"2} + Q_i^{"2})(R_i^2 + X_i^2)/V_m^2} \tag{2}$$

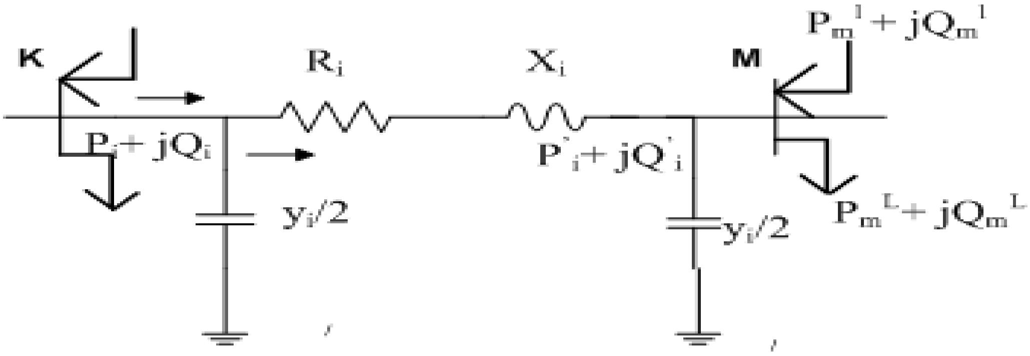


Fig 2.2: π circuit model of the distribution link

2.3 Algorithm For Proposed Method

The computational steps involved in solving the load flow problem of a single source network, by the proposed method,[2] are given in the following:

(i) Read the system data. Construct the tree and number the branches. Assume the initial voltage of all buses except the root bus.

(ii) Order the buses and compute the reduced bus impedance matrix $[Z_{red}]$. Assume the initial value of power injection at the LBPs.

(iii) Compute the active and reactive power flow through each branch of the tree from eqns.1, respectively. The power flow should be calculated in a backward direction .

(iv) Compute the voltage magnitude at the receiving end bus of each branch using eqn. 2. The voltage should be calculated in a forward direction.

(v) Compute the angle of the voltage at the end of each branch in a forward direction. Find the voltage differences at the LBPs. Update the active and reactive power injections at the LBPs using eqns. 11.

(vi) Repeat steps (iii) to (v) until the algorithm converges with an acceptable tolerance. The algorithm described above is for a mesh network. For a radial network, steps (ii) and (v) can be dropped because of the absence of LBPs and the algorithm then becomes very simple.

Finding Power Injections At LBPs

In the above mentioned algorithm finding the power injected forms a very vital part. In each iteration the voltage difference across LBP is found out which in turn gives the current and thus the power injected using reduced order impedance matrix, Z_{red} . The order of Z_{red} is same as the number of loops. The node equation is expressed as

$$[I] = [V][Y] \tag{3}$$

It is to be noted that trees are categorized in three categories. The root bus is not included in eqn.3 because it is connected to the reference bus through a negligible (or zero) impedance. The loads in the system are replaced by constant shunt admittances at a nominal voltage of l.0pu. Since the current injection to the third set (set c) is zero they are eliminated by *Kron Reduction.*

$$\begin{bmatrix} I_a \\ I_b \end{bmatrix} = \begin{bmatrix} Y_{aa} & Y_{ab} \\ Y_{ba} & Y_{bb} \end{bmatrix} \begin{bmatrix} V_a \\ V_b \end{bmatrix} \tag{4}$$

⇨
$$\begin{bmatrix} V_a \\ V_b \end{bmatrix} = \begin{bmatrix} Z_{aa} & Z_{ab} \\ Z_{ba} & Z_{bb} \end{bmatrix} \begin{bmatrix} I_a \\ I_b \end{bmatrix} \tag{5}$$

Now the voltage difference across the node can be found by the following equation.

$$[Vab] = [Va - Vb] = [Zaa - Zba][Ia] + [Zab - Zbb][Ib] \tag{6}$$

Knowing that current at LBP is equal and opposite we get

$$[V_{ab}] = [Zaa - Zba - Z_{ab} + Z_{bb}][I_a] \tag{7}$$

$$=[Z_{red}][I_a] \tag{8}$$

Since the eqn is linear so holds good for incremental values, we get

$$[\Delta V_{ab}] = [Z_{red}][\Delta I_a] \tag{9}$$

Knowing the value of ΔV_{ab} in each iteration we can calculate the ΔI_a value. Once ΔI_a is known, the incremental change in complex power injection at the first set of buses (seta) can be written as

$$[\Delta S_a] = [V_a][\Delta I_a]^* \tag{10}$$

At the end of each iteration, the active and reactive power injections at the LBPs can be updated as:

$$\begin{aligned} P_{p+1}^{I} &= P_{p}^{I} + \mathrm{Re}(\Delta S_a) \\ Q_{p+1}^{I} &= Q_{p}^{I} + \mathrm{Im}(\Delta S_a) \end{aligned} \tag{11}$$

2.4 The Problem Specific To 33 KV Line At NIT Rourkela

Layout Of the 33KV Distribution System

The 33 KV ring main system has 9 nodes (or substation) and the general technical specification has already been mentioned before. The single line diagram of the same has been drawn in the below diagram. XLPE and ACSR rabbit are used as distribution medium, The buses at substation are made of ACSR dog. Their electrical characteristics are mentioned later.

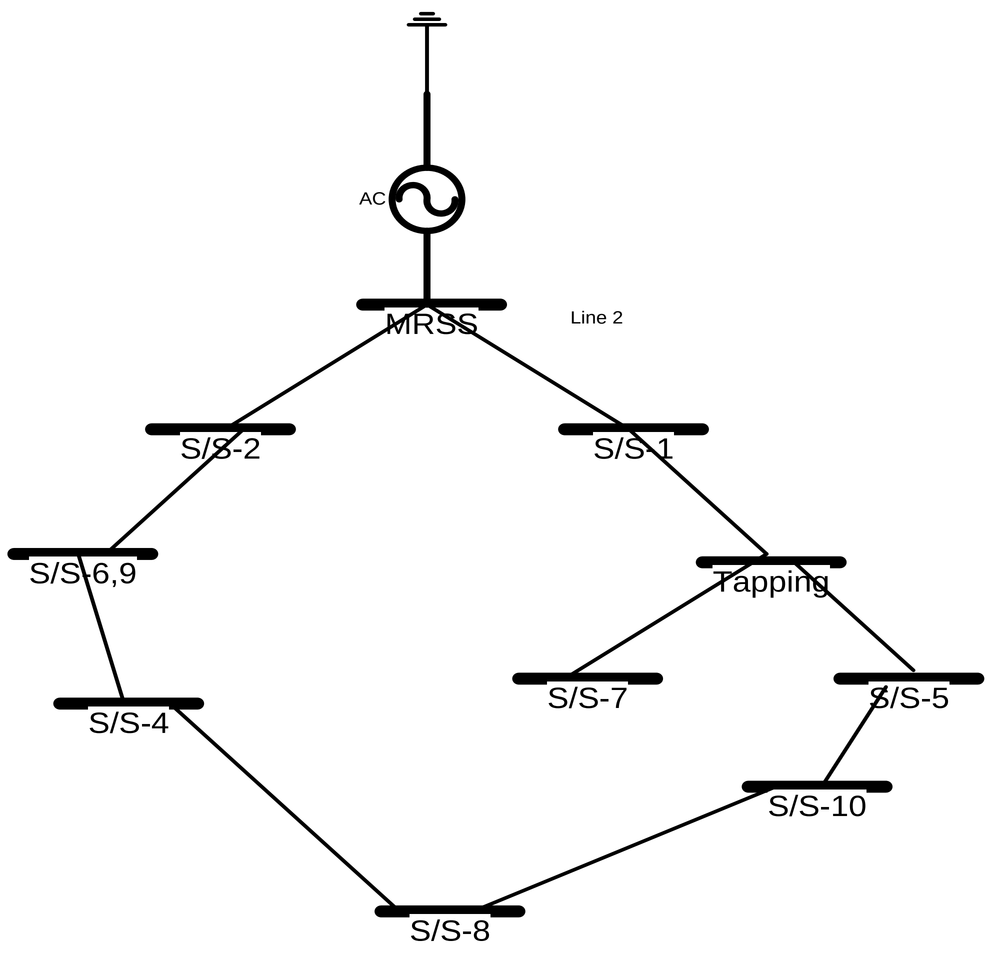


Fig 2.3: Ring Main System of NIT Rourkela 33KV line

The loop is broken at substation 8 and a dummy bus is introduced i:e h'. . The layout now represents a radial network. In short, the network with 8 buses is reconfigured to 9 buses by adding a dummy bus g'. But the behavior of the original network is preserved by injecting complex power at node g and g'. The buses are numbered according to the method discussed before. The branch numbering process starts at the first layer. The numbering of branches in any layer starts only after numbering all the branches in the previous layer. The LBPs identified in the tree construction process are h-h'. On ordering the bus we have bus h in set a, h' in set b and a,b,c,d,e,f,g,i in set C.

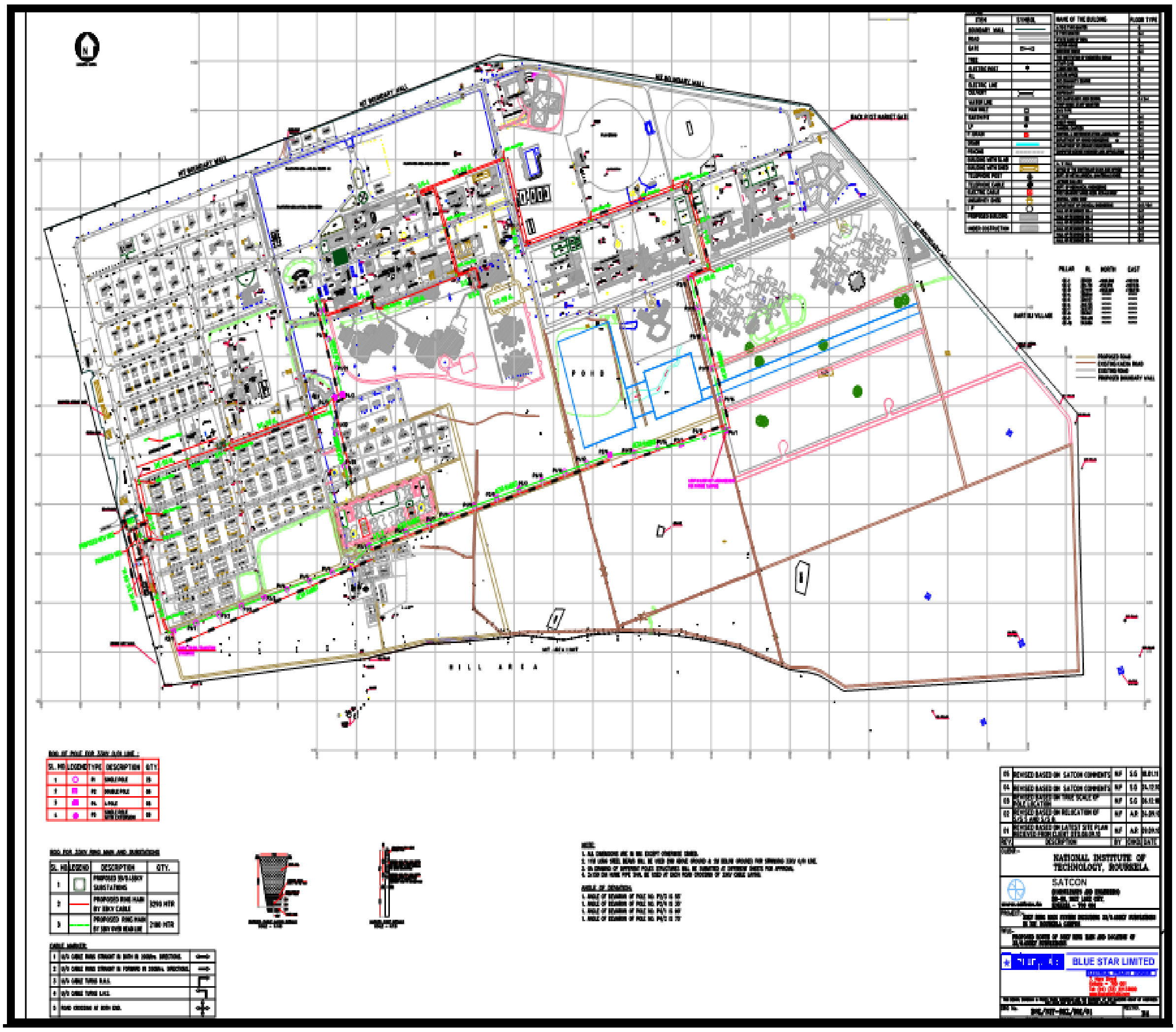


Fig 2.4: Electrical Layout of the 33KV ring main system. (Source: SATCON)

Finding the Lumped Characteristics Of the Links

The below table shows the electrical characteristics of the conducting materials that have been sanctioned by the institute.

Conductor	Resistance(Ω/Km)	Reactance(Ω/Km)	Capacitance(μfarad/Km)
XLPE	3.94	0.08	0.16
ACSR rabbit	0.555	nil	Nil

Table 2.1: Electrical Characteristics of the Conductors Used.

Having known the electrical characteristics per Km the length of each links was found out from the layout map. And the table shows the same.

Link no.	Length of XLPE(in mm)	Length of ACSR(in mm)
1	265,286	564,000
2	376,500	NA
3	1,225,000	168,000
4	272,000	NA
5	916,000	NA
6	NA	172,000
7	NA	198,000
8	318,000	NA
9	171,500	NA
10	554,531	NA

Table 2.2: Geometrical Length of Each Link.

Having known the length of each link in respective categories, and their electrical characteristics we can calculate the lumped electrical impedance and admittance.

Line no.	From bus	To bus	R(Ω)	X(Ω)	y/2(mho)	
1	0	1	0.0037	5.84E-05	0.00484	
2	0	2	0.0041	8.29E-05	0.00686	
3	1	3	0.0037	8.65E-05	0.003061	
4	2	4	0.003	6.01E-05	0.00498	
5	3	5	0.0099	0.000404	0.01671	
6	4	6	0.0003	0	0	
7	4	7	0.0003	0	0	
8	5	8	0.0035	0.00014	0.005804	
9	7	9	0.0019	7.55E-05	0.003124	

Table2.3: Electrical Characteristics of Each Link.

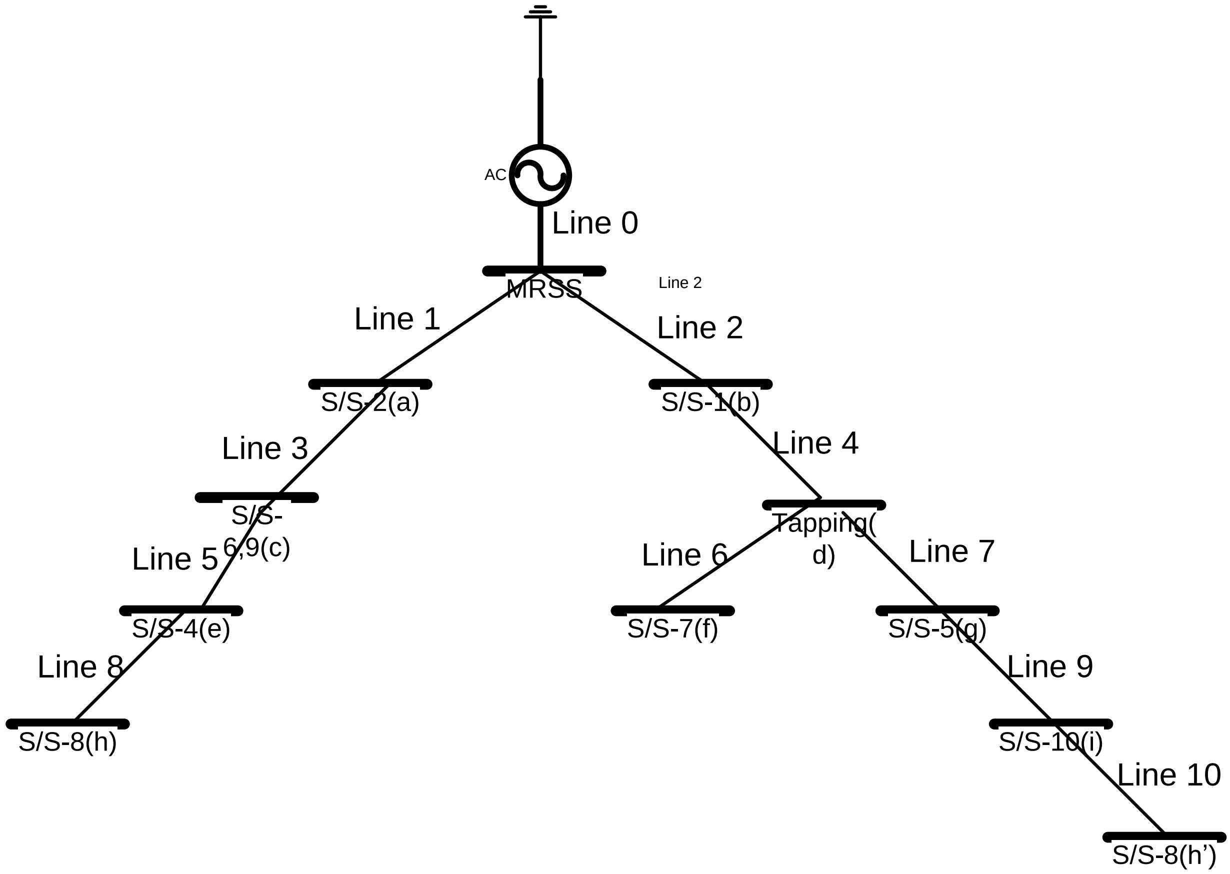


Fig 2.5: Ring Main System Made Radial to Solve Load Flow Analysis.

Finding Power Injection at LBP

Here it involves the task of finding Z_{red} .First the admittance matrix is found out by replacing the load at each node with equivalent admittance. The inverse of [Y] thus found gives impedance matrix and the Z_{red} . Then following equation 11 we find the injection power in each iteration. The MATLAB function to find out the same is as follows:

```
function [dSa]=injectpower(dVab,Va)
Zred=-0.0804 - 0.0070i;
dIa=dVab/Zred;
dSa=Va*(conj(dIa));
end
```

2.5 Results

With the tolerance of 0.001 we stop the iteration and current in each line is found out. And load flow results found, are shown below

\#

Loadflow Analysis of the 33KV Main Line

Bus No	V pu	Angle Degree	Injection KW	Injection KVar	Load KW	Load KVar
1	0.9953	-0.2027	0.000	0.000	426.000	225.000
2	0.9988	-0.0320	0.000	0.000	426.000	225.000
3	0.9910	-0.3906	0.000	0.000	1704.000	900.000
4	0.9984	-0.0440	0.000	0.000	0.000	0.000
5	0.9854	-0.7321	0.000	0.000	852.000	450.000
6	0.9983	-0.0453	0.000	0.000	426.000	225.000
7	0.9977	-0.0628	0.000	0.000	1275.000	790.500
8	0.9845	-0.8274	1831.500	159.459	2556.000	1578.000
9	0.9940	-0.1458	0.000	0.000	1275.000	790.500
10	0.9850	-0.3265	-1831.500	-159.459	2556.000	1578.000

Table 2.4: Results of the Load Flow Analysis of the 33 KV Line

And the Line Losses for each line calculated are as follows:

Line Losses

Line no	Line losses KW	Line no	Line Losses KW
1	28.094	6	0.023
2	1.242	7	6.017
3	22.851	8	3.025
4	0.230	9	24.879
5	19.827	10	45.801

Table 2.5: Calculated Line Losses for the 33 KV Line

2.6 Conclusion

An efficient load flow method for a distribution system has been developed without ignoring the shunt admittances. The application of the proposed method to a radial network is very simple and straightforward. However, for a mesh network, the network should be converted to an equivalent radial configuration by breaking the loops. The conversion process added some dummy buses in the network. The power injections at the loop break points are computed by using a reduced order bus impedance matrix. The order of the matrix is the same as the number of loops in the original network. The effects of both the load and shunt admittances are considered in the impedance matrix. Because of the incorporation of shunt admittances, the proposed method can also be used to solve the load flow problem of a single source transmission system.

Chapter 3

Data Acquisition System (DAS)

The most significant component of the plan of incorporating intelligent algorithms to improve reliability of a distribution system is the Data Acquisition System (DAS). This is the process of obtaining real-time data from the system while it operates. The data collected can then be used to monitor and analyze the system parameters, system health and devise suitable mechanism for imparting autonomous intelligence to the system.

3.1 DAS Architecture

All Data Acquisition Systems, invariably, comprise of the following subsystems:

- *Data Collecting Unit (Multi Function Meters)*
- *Data Conditioning Unit (MODEM)*
- *Transmitting Unit (Antenna and Communication Protocols)*
- *Server Storage Unit(Database and Web server)*

In the 33-KV Ring Main System, the Data Acquisition System has the objective of collecting real-time data from the Substations and relaying it to the *Master Control Server.* The data at the server can then be monitored by users using simple web browser based application. All the Feeder meters installed in the Substation will have RS 485 ports to communicate the data to the transmitting antenna. The *Multi Function Meters (MFMs)* shall be connected through daisy chain link through RS 485 ports over MODBUS protocol and finally connected to their station Remote Terminal Unit (RTU). Each RTU shall be connected to the GPRS/GSM Modem, which shall be connected to the DAS server through GSM Network. The communication between the RTUs at Substations and Master Server shall take place over the IEC 60870-5-104. The communication protocol and the interfacing between DAS Server and NIT Server shall be based upon the industry standard open protocol viz. MODBUS, OPC.

The RTUs shall have the requisite number of I/O modules to interface with the direct I/Os from the breakers. From the Multi Function Meters, real time Analog values of Active Power, Reactive Power, Current, Voltage and Frequency, Power factor will be obtained periodically. In addition, health status, synchronization and sensing status data is also to be included in a transmission capsule which is to be forwarded to the Master DAS Server. All input data received shall be checked for reasonability and rejected, if found unreasonable.

The data once accumulated in the server will be stored in standard SQL databases and will be available for future use. The data can be accessed by using any web based client using a standard web browser. The data can be put into analysis under various algorithms and suitable interpretations be made.

3.2 Data Analysis

The Real-Time metering data obtained from the various RTUs connected all over the campus is collected in a secure Master Server database. Various Artificial Intelligence Techniques and Learning algorithms can be successfully tested and implemented once we have the data set available.

In this project work, we have used the *Multi Layer Perceptron* based Artificial Neural Networks to forecast the future load on the Distribution System network and, then accordingly implement the Demand-Side Tariff Management System. Finally, we also have developed a web portal: *NITR e-Power Management System (e-PMS)* specially dedicated to host the data collected by the DAS and its analysis.

Data Storage and Access:

The data collected from the various RTUs will be stored securely in a Database physically located on the Master Server. The database can be accessed at any time anywhere inside the campus, in a local hosted web server. Restricted access can be provided to all clients enabling them to access and analyze the data using a standard web browser.

Demand-Side Tariff Management Systems:

The data available from the RTUs can also be utilized in studying various tariff management systems, particularly the Demand Based Tariff management system. The Demand based tariff transforms the flat income curve of the utility to a more complex peak load dependency income profile. As a result, the utility tends to gain proper monetary benefits and the net energy consumption is also reduced. With the available transformer parameters from the RTUs, we would be able to simulate the tariff management systems and study the various economical optimization techniques.

Apart from the above mentioned applications, the data collected from the DAS can also be used to evaluate the Distribution Transformer Losses by Load Monitoring Method. After Fault Load Flow calculation and optimal mitigation can also be performed once we have an access to the database.

Chapter 4

Artificial Neural Networks (ANN) Approach

4.1 Introduction

The purpose of mathematical modeling is to come up with a set of equations that describe the interrelations between the system parameters. An equation can be formed from algebraic, differential, integral, difference, or functional equations. If mathematical modeling of a system is not feasible, one looks to come up with different analytical models. Such models are designed by solving two cardinal problems in modern science and engineering:

1) Learning from experimental data by neural networks
2) Embedding existing structured human knowledge into workable mathematics by fuzzy logic models

The above two models, i.e. neural network and fuzzy logic models are the most important constituents of soft computing. **Soft Computing** is a field within computer science which uses inexact solutions to compute hard tasks (such as the solution of NP-complete problems, for which an exact solution cannot be derived in polynomial time). [4]

An *Artificial Neural Network* is a device that is designed to model the way in which the human brain performs various tasks. The network is implemented by using electronic components or is simulated in software on a digital computer. A neural network is a massively parallel distributed processor made up of simple processing units, which has a natural propensity for storing experimental knowledge and making it available for use. It resembles the brain in two respects:

1) Knowledge is acquired by the network from its environment through a learning process.

2) Inter-neuron connection strengths, known as *synaptic weights*, are used to store the acquired knowledge.

The procedure used during the learning process is called a *Learning Algorithm.* This algorithm is used to modify the weights of the network in an orderly fashion to obtain a desired design objective. [5]

4.2 History of ANN

Research in ANN was inspired by the desire to come up with artificial systems that are capable of solving various problems much the same way a human brain would solve it. The first significant research on neural networks was published in 1943 by Warren McCulloh and Walter Pitts. They came up with a simple neuron model and implemented it as an electrical circuit. In 1949, Donald Hebb was the first to point out the connection between psychology and physiology, pointing out that a neural network becomes stronger with every time it is used. Technological advancements in computers in subsequent years made it possible to simulate and test theories about artificial neural networks. Perceptron was developed by Frank Rosenblatt in 1958. [4]

After an initial period of enthusiasm when the capabilities of neural networks were exaggerated beyond proportions, the evolution of neural networks went through a lackluster period especially in 1969 when Minsky and Papert published ‘Perceptrons’, condemning Rosenblatt’s perceptron. However, through persistent efforts of scientists like Teuvo

Kohonen and Stephen Grossberg, new breakthroughs were made. John Hopfield introduced the recurrent type neural network in 1982. Following this, the back-propagation learning algorithm was developed and neural network advancement received a boost.

As research continues, more and more types of networks are being introduced, although less emphasis is being placed on the connection to biological networks.

4.3 Need for ANN

Neural networks are very adept in extracting meaning from complicated or imprecise data. They can be used to detect patterns and trends that are too complex to be noticed by either humans or other computer techniques. A neural network that has been trained can be compared with an "expert" in the problem it has been given to analyze. This “expert” can then be used to provide accurate projections given new situations and answer "what if" questions.

Other advantages include:

1) **Adaptive learning:** It is the ability of neural networks to learn to perform tasks based on past training.

2) **Self-Organization:** An ANN is capable of organizing or representing information it receives during learning time on its own.

3) **Real Time Operation:** It is possible to carry out ANN computations in parallel, and special hardware devices are being designed to take advantage of this capability.

4.4 Benefits of ANN

1) They are very powerful computational devices.
2) They are very efficient because of their capability to handle massive parallelism.
3) There is no n need for complex programs as they can learn from the training data itself.
4) They are highly fault tolerant.
5) They have high noise tolerance.

4.5 Mathematical Model of a Neuron

A neuron is the fundamental element of a neural network. It is the information processing unit of the network. The three basic elements of the neuron model are:

1) A set of weights, each with a strength of its own. A signal x_j connected to k^{th} neuron is multiplied by the weight w_{kj}. The weight can take both positive and negative values.
2) An summer for adding the input signals, pre-weighted by their respective weights
3) An activation function for limiting the amplitude of the output of a neuron. It is also known as squashing function which squashes the amplitude range of the output signal to some pre-defined finite value. [5]

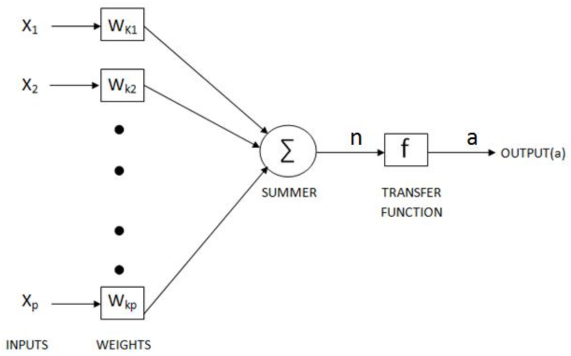
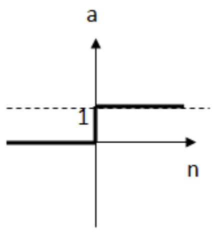


Figure 4.1: Model of a Neuron

For the shown neuron model, we have: $\mathbf{v_k = \sum^{p}_{j=1} w_{kj} x_j}$ and $\mathbf{y_k = \varphi(v_k + \theta_k)}$

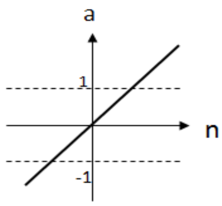
Transfer Functions

1) Hard-Limit Transfer Function



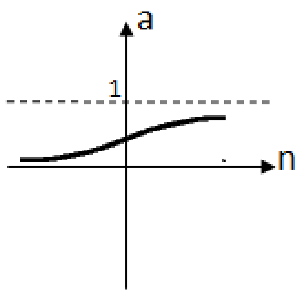
The hard-limit transfer function limits the output of the neuron to either 0, if the input argument *n* is less than 0, or 1, if *n* is more than or equal to 0.

2) Linear Transfer Function



Neurons of this type are used as linear approximators in Linear Filters.

3) Log-Sigmoid Transfer Function



The log-sigmoid transfer function is commonly used in back-propagation networks, mostly because it is differentiable.

Network Architectures

There are three fundamental different classes of network architectures:

1) **Single-layer Feed forward Networks :**

The neurons are organized in the form of layers. In the simplest form of a single-layered network, there is an input layer of source nodes that projects directly onto an output layer of neurons, but not vice versa. This network is strictly a Feed forward type. In this type of network, there is only one input and one output layer. Input layer is not considered as a layer since no mathematical calculations take place at this layer.

2) **Multilayer Feed forward Networks:**

This type of neural network consists of one or more hidden layers. The corresponding nodes are called hidden neurons, the function of which is to usefully modify the external input so that the network output reaches the desired value. By adding more hidden layers, the network is enabled to extract higher order statistics. The input signal is applied to the second layer neurons. Its output is used as input to the next layer and so on.

3) **Recurrent networks :**

A recurrent neural network has at least one feedback loop. It may consist of a single layer of neurons with each neuron feeding its output signal back to the inputs of all the other neurons. Self-feedback is the case when the output of a neuron is fed back to its own input. The feedback loop greatly enhances the learning capability of the neural network, enhancing its performance.

4.6 Learning Processes

Learning process implies a procedure for modifying the weights and biases of a network. Its purpose is to train the network to solve a problem or perform a task. They fall into three broad categories:

1) **Supervised Learning:**

A pre-defined set of training data that reflect the network behavior are used in this type of learning. As the inputs are applied to the network, the network outputs are compared to the targets. The learning rule then modifies the weights and biases of the network so that the network outputs are closer to the target outputs.

2) **Reinforcement Learning:**

It is similar to supervised learning. However, here instead of being providing the correct output for each network input, the algorithm is given a grade. The grade is a measure of the network performance over some sequence of inputs.

3) **Unsupervised Learning:**

The layer weights and biases are updated in response to network inputs only. No target outputs are specified. These algorithms make use of some kind of clustering operation. They learn to group different input patterns into a finite number of classes.

4.7 Back-Propagation Algorithm

Multiple layer perceptrons have been applied successfully to solve some difficult diverse problems by training them in a supervised manner with a highly popular algorithm known as the error back-propagation algorithm. This algorithm is based on the error-correction learning rule. It may be viewed as a generalization of an equally popular adaptive filtering algorithm- the *Least Mean Square* (LMS) algorithm. Error back-propagation learning comprises of two computation phases through the different layers of the network: a forward computation and a backward computation. In the forward pass, an input vector is applied to the nodes of the network, and its effect percolates through the network to give the final layer output which is the net output of the system. Finally, a set of outputs is produced as the actual response of the network. During the forward pass the weights of the networks are all fixed. During the backward pass, the weights are all adjusted in accordance with an error correction rule. The actual response of the network is subtracted from a desired response to produce an error signal. This error signal is then passed backwards through the network, running in opposition to the direction of synaptic connections. The weights are adjusted to make the actual response of the network move closer to the desired response. A multilayer perceptron has three distinctive characteristics: [6]

1. The model of each neuron in the network includes a nonlinear activation function. The sigmoid function is commonly used which is defined by the logistic function:

$$y = 1/\ 1+\exp(-x) \qquad (3.1)$$

Another commonly used function is hyperbolic tangent

$$y = 1-\exp(-x)/1+\exp(-x) \qquad (3.2)$$

The presence of nonlinearities is important because otherwise the input- output relation of the network could be reduced to that of single layer perceptron.

2. The network contains one or more layers of hidden neurons that are not part of the input or output of the network. These hidden neurons enable the network to learn complex tasks.

3. The network exhibits a high degree of connectivity. A change in the connectivity of the network requires a change in the population of their weights.

Thus, owing to their vast non-linear parallel processing capabilities, neural networks have become the most widely popular soft computing technique.

Chapter 5

Study and Analysis of Short-Term Load Forecasting

5.1 Introduction

"An estimate of power demand at some future period is known as load forecasting".

Load forecasting is a vital component for energy management system. Load forecasting helps making important decisions such as on purchasing and generating power, infrastructure development and load switching. In addition to reducing the generation cost, it also helps in reliability of power systems. Load forecasting is also important for planning and operational decision conducted by electric utility companies. With changes in weather conditions and supply and demand fluctuating and energy prices increasing at a very a high rate during peak situations, load forecasting is vitally important for utilities. [7]

5.2 Types of Load Forecasting

In terms of lead time, load forecasting is divided into four categories:

1) *Long-term forecasting* - lead time of more than one year
2) *Mid-term forecasting* - lead time of one week to one year
3) *Short-term load forecasting* - lead time of 1 to 168 hours
4) *Very short-term load forecasting* - lead time less than one day

The forecasts of different time horizons are highly important for various operations of a utility company. The natures of the forecasts are also different. The system operators use these load forecasting result as the basis of off-line network analysis in order to determine if the system is vulnerable. If so, corrective actions need to be prepared, like load shedding, power purchase etc. For example, for a particular region, next day prediction of load is possible with an accuracy of around 1-3%. However, it is impossible to predict the next year peak load with the similar accuracy since accurate long-term weather forecasts are not available. Since in power systems the next days' power generation must be scheduled every day, day ahead **Short-Term Load Forecasting (STLF)** is a necessary daily task. Its accuracy affects the reliability and economic operation of the system to a large extent. Under prediction of STLF leads to insufficient reserve capacity preparation and over prediction leads to the unnecessarily large reserve capacity.

5.3 Important Factors Affecting Forecast

For short-term load forecasting several factors should be considered, such as time factors, weather data etc. From the observation of the load curves it can be seen that there are certain rules of the load variation with the time point of the day.

Weather conditions also influence the load. Forecasted weather parameters are the most important factors in short-term load forecasts. Various weather variables are considered for load forecasting. However, Temperature and humidity are the most commonly used load predictors. Among the weather variables, two composite weather variables functions, the THI (temperature-humidity index) is broadly used by utility companies. THI is a measure of summer heat discomfort. Most of the electric utilities serve different types of customers such as residential, commercial, industrial etc. Though the electric usage pattern differs for customers belonging to different classes, it is somewhat same for customers in each class. Therefore, most utilities distinguish load behaviour on a class by-class basis. The system load is the sum of all the consumers' load at the same time. [8]

Various factors influencing the system load behavior, can be classified into the following major categories

● Weather

● Time

● Random disturbance

Weather:

Weather factors include temperature, humidity, cloud cover, light intensity, wind speed etc. The change in the weather causes the change of consumers' usage of appliances such as heaters and conditioner. Temperatures of the previous days also affect the load profile. Humidity is also an important factor, because it affects the human being's comfort feeling greatly. That's why **temperature-humidity index (THI)** is the most effective tool employed in load forecasting.

Time:

The time factors include the time of the year, the day of the week, and the hour of the day. There are significant differences in load between weekdays and weekends. The load on different weekdays also behaves differently. This is particularly true during summer. Holidays are far more difficult to forecast than non-holidays because of their relative infrequent occurrence.

Random Disturbance:

The modern power system is composed of numerous electricity users. Although it is not possible to predict how each individual user consumes the energy, the amount of the total loads of all the small users shows good statistical results leading to smooth load curves. But there are always some random disturbances like the start up and shutdown of the large loads leading to a sudden impulse in the load curve. The start up and shutdown time of these users is quite random and when the data from such a load curve are used in load forecasting training, the impulse component adds to the difficulty of load forecasting. Certain special events, which are known in advance but there effect on load is not certain, are also a source of random disturbance. An example of a special event is, a world cup cricket match, which the operators know, will increase usage of television, but cannot decide the amount of the usage.

5.4 Forecasting Techniques

Different forecasting techniques serve different purposes. Since we are concerned with forecasting the load for the next day, we use the **Short Term Load Forecasting** technique. The research approaches of short-term load forecasting can be mainly divided into two categories: *Statistical Methods* and *Artificial Intelligence* methods. [9] The statistical category includes multiple linear regression [10], stochastic time series [11], general exponential smoothing [12], state space [13], etc. Usually statistical methods can predict the load curve of ordinary days very well, but they lack the ability to analyze the load property of holidays and other anomalous days, due to the inflexibility of their structure. Expert system [14], artificial neural network (ANN) [15], fuzzy inference [16], and evolutionary algorithm belong to the computational intelligence category. Usually statistical methods predict the load curve of ordinary days very well, but are unable to analyze the load property of holidays, due to the inflexibility of their structure. Artificial Neural Network is good in dealing with the nonlinear relationship between the load and its relative factors, but the shortcoming lies in long training time and over fitting

Some methods used to implement Short-term load forecasting are described below:

1) <u>Regression Methods:</u>

Regression is one of most widely used statistical techniques. Feinberg *et al.* ([17], [18]) developed a statistical model that learns the load model parameters from the historical data. Feinberg *et al.*([17], [18]) studied load data sets provided by a utility company in North eastern US. Several load models were compared and was concluded that the following multiplicative model shown below is the most accurate

$$L(t) = F(d(t), h(t)) \cdot f(w(t)) + R(t),$$

where $L(t)$ is the actual load at time t, $d(t)$ is the day of the week, $h(t)$ is the hour of the day, $F(d, h)$ is the daily and hourly component, $w(t)$ is the weather data that include the temperature and humidity, $f(w)$ is the weather factor, and $R(t)$ is a random error. To estimate the weather factor $f(w)$, regression model was used:

$$f(w) = \beta 0 + \beta j\ Xj\ ,$$

where Xj are explanatory variables which are nonlinear functions of current and past weather parameters and $\beta 0$, βj are the regression coefficients. The parameters of the model can be calculated iteratively. Start with $F = 1$ and then use the above regression model to estimate f. Then estimate F, and so on.

2) <u>Time Series:</u>

Time series methods are based on the assumption that the data have an autocorrelation, trend or seasonal variation. The methods detect and explore such a structure. ARMA (Autoregressive Moving Average), ARIMA (Autoregressive Integrated Moving Average) and ARIMAX (Autoregressive Integrated Moving Average with Exogenous Variables) are the most often used classical time series methods. ARMA models are generally used for stationary processes while ARIMA is an extension of ARMA to non stationary processes **[25].** Fan and McDonald [19] and Cho *et al.* [20] describe implementations of ARIMAX models for load forecasting.

3) <u>Neural Networks:</u>

Artificial neural networks (ANN or simply NN) have been a widely studied load forecasting technique. [21] Neural networks are essentially non-linear circuits that have the capability to do non-linear computations. The outputs of an artificial neural network are some linear or non-linear mathematical function of its inputs. The inputs may be the outputs of other network elements as well as actual network inputs. Feedback paths are also used sometimes. The most widely used artificial neural network architecture for load forecasting is ***back***

propagation. This network uses continuously valued functions and supervised learning. Artificial neural networks with unsupervised learning do not require pre-operational training.

Medium and Long Term Forecasting

Two of the methods, **End-use and Econometric approach** are broadly used for medium- and long-term forecasting. The end-use modelling, econometric modelling, and their combinations are the most often used methods for medium- and long-term load forecasting. Long-term forecasts include the forecasts on the population changes, industrial construction, economic development, and technology development.

End-use models End-use models focus on the various uses of electricity in the residential, commercial, and industrial sector. These models are based on the principle that electricity demand is derived from customer's demand for light, cooling, heating, refrigeration, etc. The end-use approach estimates energy consumption by using information on end use and end users, like appliances, sizes of houses, the customer use, their age etc.

Econometric models The econometric approach combines economic theory and statistical techniques for forecasting electricity demand. The approach estimates the relationships between energy consumption (dependent variables) and factors influencing consumption **[25]**. Integration of the econometric approach into the end-use approach introduces behavioural components into the end-use equations. The relationships are estimated by the least-squares method or time series methods.

5.5 Approach for Short Term Load Forecast

As discussed above, a broad spectrum of factors affect the system's load level such as trend effects, weather effects, random effects like human activities, load management and thunderstorms. Hence the load profile is dynamic in nature with temporal, seasonal and annual variations. In the present project, a system was developed that predicted 24-hour ahead load demand. In this system load demand/consumption of 48 half hours for the past 30 days was taken as the load demand input and daily **THI** (Temperature Humidity Index), taken as weather input is calculated as:

$$\mathbf{THI = [(T_{max} + T_{min}) * (R_{max} + R_{min})/4] * 0.0025}$$

where T_{max} and T_{min}, R_{max} and R_{min} are the maximum and minimum temperature and Relative Humidity (in percent) recorded for the given day.

Load demand and weather data of New South Wales, Australia was used as input parameters. The inputs were fed into the Artificial Neural Network (ANN) and after sufficient training were used to predict the load demand for the next day. A schematic model of system is shown in **Fig.5.1**

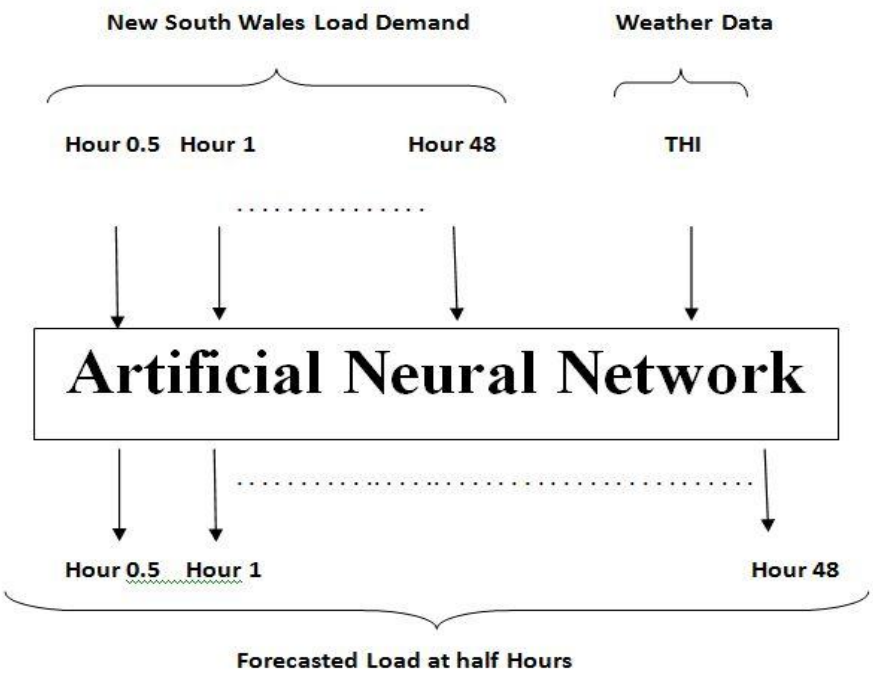


Fig 5.1: Input-Output Schematic for load forecasting

For the present load forecast, two approaches were followed, both involving Neural Network model. One was using the *MATLAB Neural Network Toolbox* and other was a *self-developed program.*

Load Forecasting Using Neural Network Toolbox

This approach used is a Feed Forward network with a single hidden layer. The number of neurons in the hidden layer was taken as 20. The activation function used in the hidden layer neuron was “Log-Sigmoid”. The output was obtained with 1 neuron using “Pure linear” activation function. The ANN was implemented using MATLAB 7.1. The training algorithm *‘Traingdx’* was used, which is an adaptive learning algorithm. *‘Traingdx’* can train any network, as long as its weight, net input, and transfer functions are derivable functions.

Back-propagation is used to calculate derivatives of performance ‘*perf’* with respect to the weight and bias variables X. Each variable is adjusted according to gradient descent with momentum. The number of epochs while training was set at 2000, which led the system to be sufficiently trained. **Fig 5.2** shows the network structure for the load forecast using neural network toolbox.

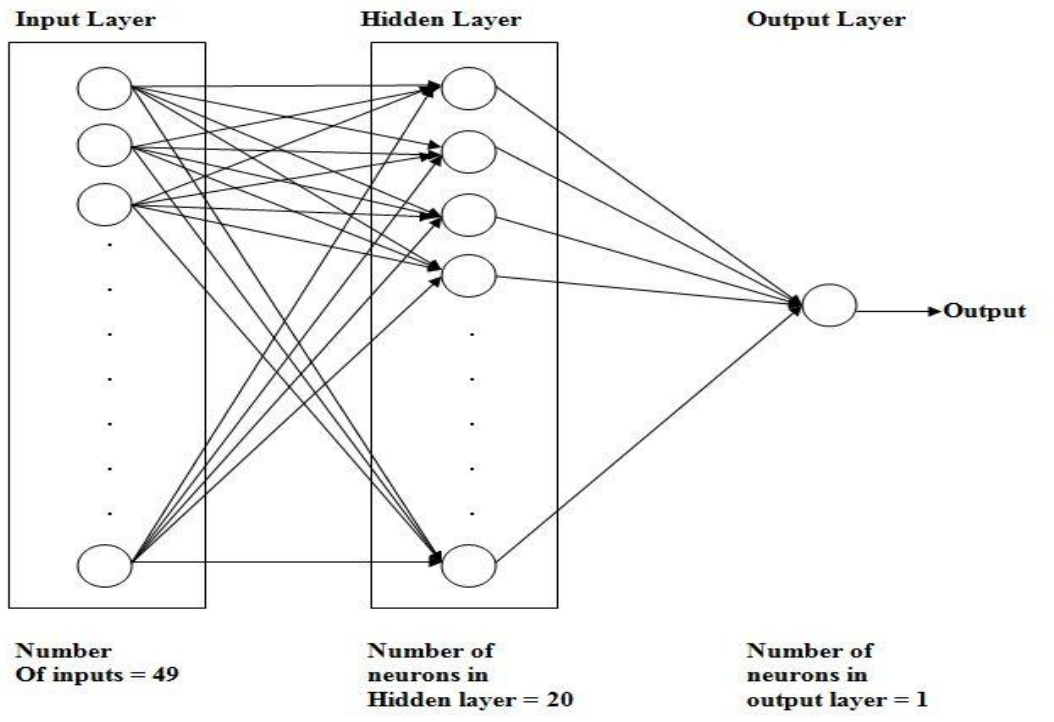


Fig 5.2: Network Structure for load forecasting

Pre-Processing and Post-Processing of Training Data

The data employed for training and testing the neural network were obtained from the *Australian Energy Market Operator (AEMO)* website for the period January-March 2010. The weather data of New South Wales, Australia was obtained from *Australian Bureau of Meteorology* for the same period. Due to wrong measurements and other human errors, some out-of- range values were observed in the historical load data. Corrections were made to such outlier values by replacing them with the average of both the preceding and succeeding values in the series. The data is initially preprocessed using *'prestd'* function to obtain data with mean as 0 and standard deviation as 1. This is followed by a principle component analysis that retains only those components which contribute more than 4 percent to the variance in the dataset. This is achieved using *'prepca'*. The normalized data were then simulated to obtain normalized outputs. The normalized outputs were processed using the *'poststd'* function to obtain de-normalized results. Subsequent new data are processed using *'trastd'* and *'trapca'* that are similar to *'prestd'* and *'prepca'*. *'trastd'* preprocesses data using a pre-calculated mean and standard deviation. *'trapca'* performs a Principal Component Analysis (PCA) similar to *'prepca'*. **Table 5.1** shows the load demand and THI of the New South Wales of the six hours for past 7 days. Similarly data for the rest of the hours of the other days were fed to the network.

Day	01-Jan	02-Jan	03-Jan	04-Jan	05-Jan	06-Jan	07-Jan
HOUR	DAY 1	DAY 2	DAY 3	DAY 4	DAY 5	DAY 6	DAY 7
0	7574.85	7574.85	7284.1	6917.57	7636.34	7878.93	7675.56
00:30	7809.31	7343.3	7053.93	6813.43	7438.78	7625.24	7433.46
01:00	7483.69	7099.73	6820.07	6633.42	7210.46	7394.02	7216.3
01:30	7117.23	6779.8	6537.47	6424.02	6892.97	7076.3	6940.48
02:00	6812.03	6497.47	6362.67	6218.66	6634.05	6810.51	6692.82
02:30	6544.33	6347.69	6178.57	6106.83	6422.19	6645.99	6477.56
03:00	6377.32	6223.64	6088.35	6067.46	6335.01	6577.41	6452.34
03:30	6282.85	6182.07	6077.74	6083.12	6313.4	6568.49	6415.95
04:00	6211.49	6140.82	6059.53	6153.59	6358.52	6611.57	6488.25
05:00	6248.31	6224.48	6101.68	6317.88	6515.13	6767.67	6627.1
06:00	6198.61	6260	6111.6	6522.17	6563.86	6897.65	6815.18
THI	16.04	9.5	7.86	5.78	4.77	5.3	3.71

Table 5.1: Load Demand and THI of New South Wales for input to the Network

5.6 Results

The results obtained from testing the trained neural network on new data for 48 half hours of a day for few arbitrary days are presented below in graphical form. Each graph shows a plot of both the “predicted” and “target” load in kW values against the half hour of the day.

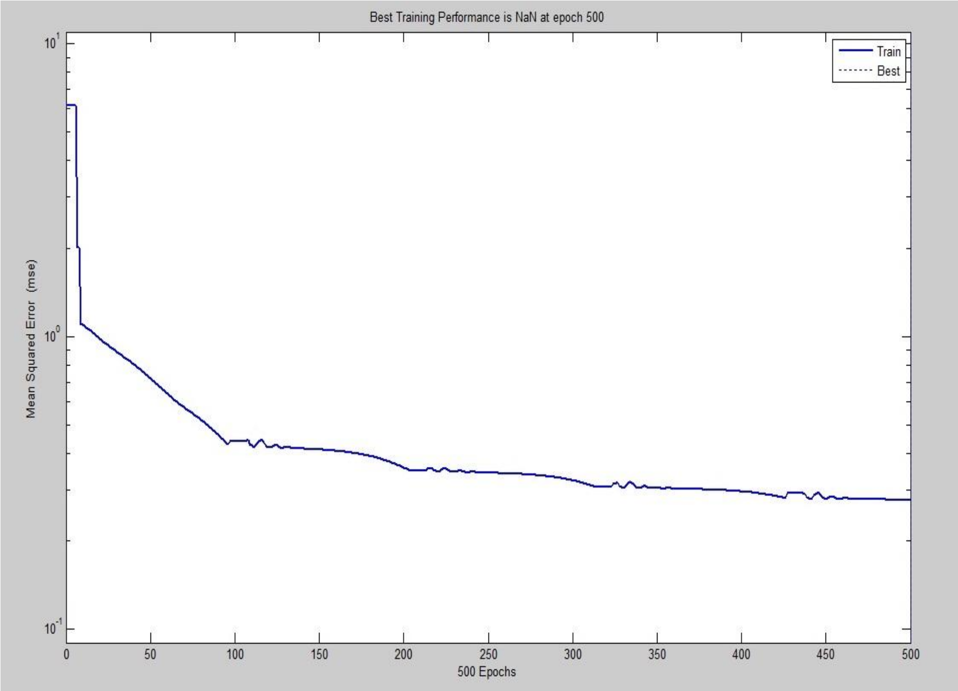


Fig 5.3: Performance Plot

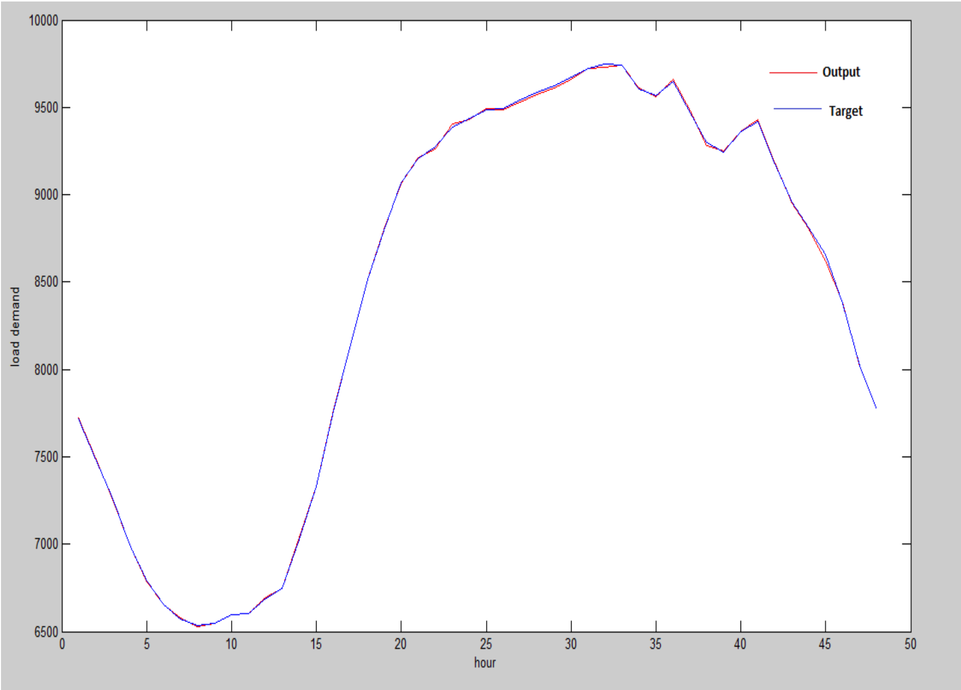


Fig 5.4: Actual v/s predicted Load for Day 1

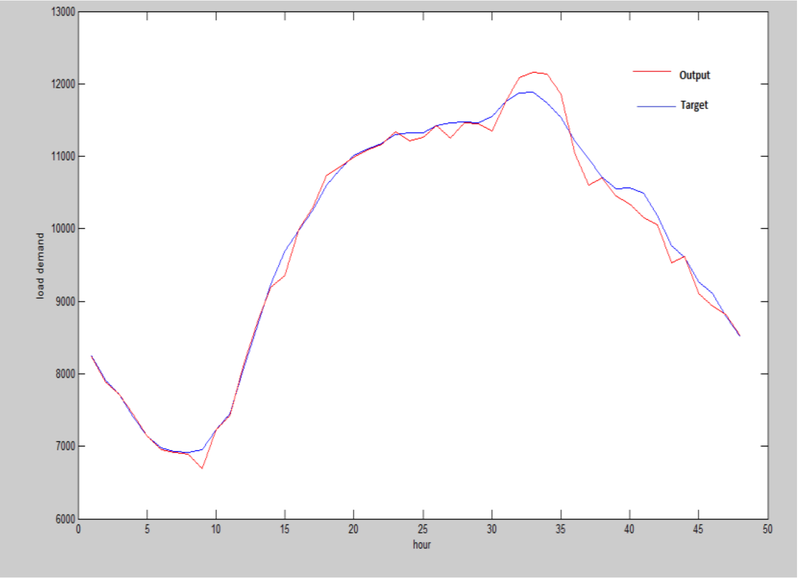


Fig 5.5: Actual v/s predicted Load for day 10

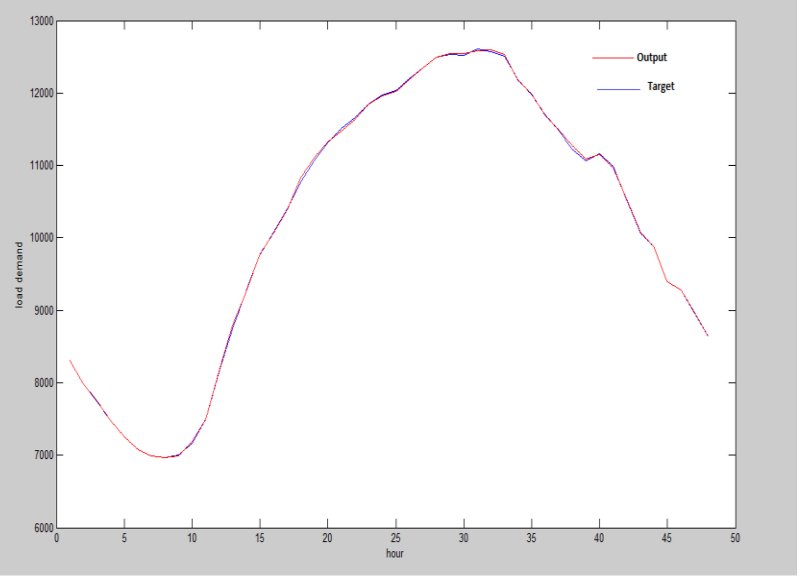


Fig 5.6: Actual v/s predicted Load for day 12

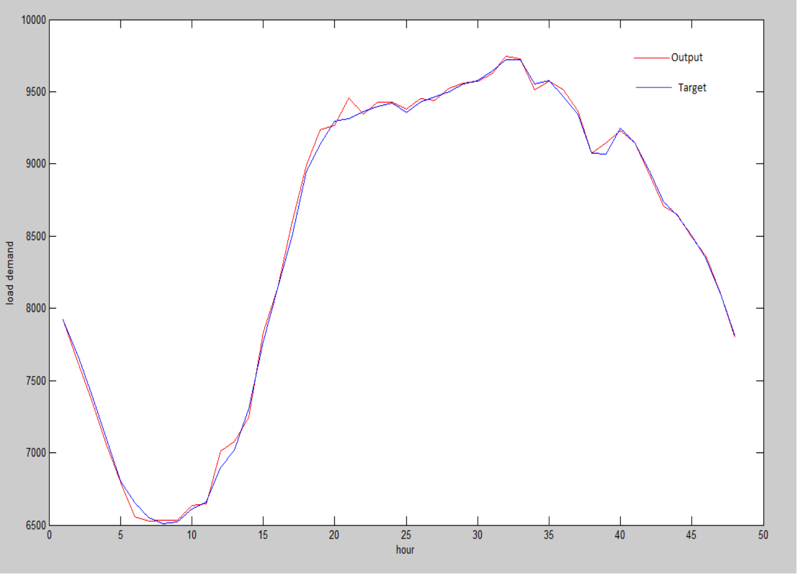


Fig 5.7: Actual v/s predicted Load for day 22

Self-Developed Program for Load Forecasting

Using the concept of *Widrow-Hoff Back propagation* based learning technique, a simple Neural Network application was developed using MATLAB as the scripting language. The program constructed is meant for short term load forecasting, that is one day ahead load forecasting. The results obtained from the load forecasting are used to define the tariff rates for the next day, which are the declared to the general public through the NITR e-PMS portal.

Employing the above mentioned techniques and methods of artificial neural network, the following program was written for short term load prediction. The inputs given to the system consists of 60 input elements where the first half elements are the load demand for the previous month at any particular instant and other half is Temperature Humidity Index **(THI)** for the past 30 days.

Network Specifications:

• No. of layers: *3 (Input layer, Hidden layer, Output layer)*

• No of neurons in hidden layer: *20*

• No of neurons in output layer: *1*

• Activation function of hidden layer: *logsig*

• Activation function of output layer: *purelin*

• Training algorithm: *Back-Propagation*

• Learning rate (α): *0.2*

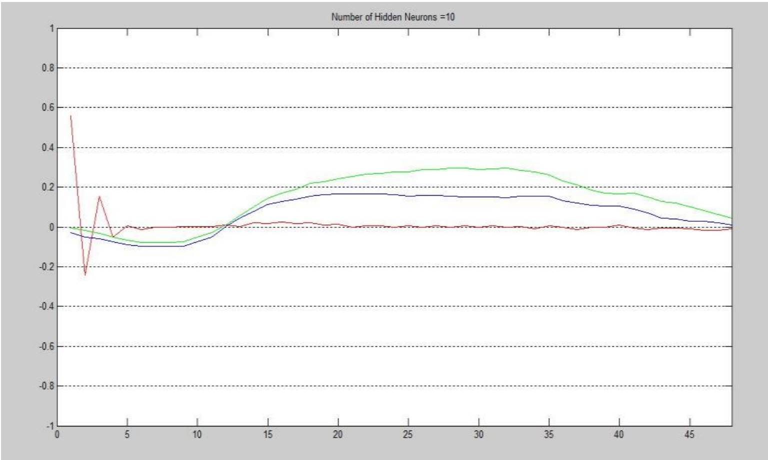
• No. of input variables: *60 (30 as load demand of previous month at that hour, 30 as* ***THI*** *for the past month)*

• No of output variable: *1 (load at particular Hour of the next day)*

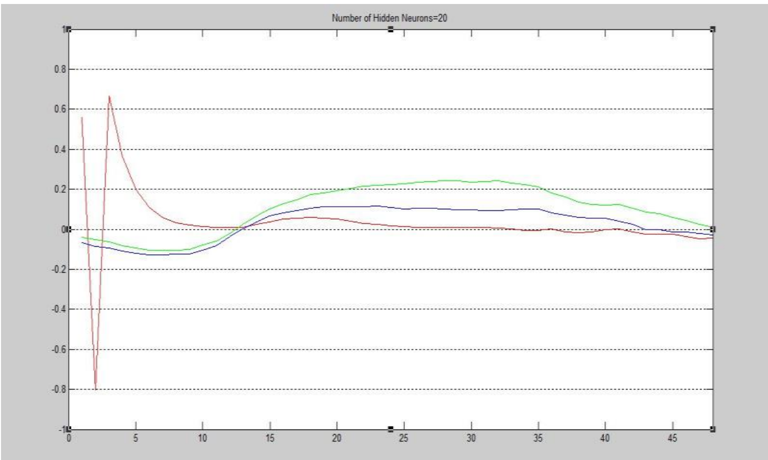
• No. of epochs for training: 20

Hidden Layer Optimization:

The number of neurons in the hidden layer was optimized after the network performance was studied taking different numbers of hidden layer neurons. It was observed that for Number of Hidden Neurons exceeding 20, there was no significant improvement in network performance rather the error overshoot kept increasing, the system complexity and execution time of the program were severely affected. Hence, the optimal number of hidden neurons is 20.



(i)



(ii)

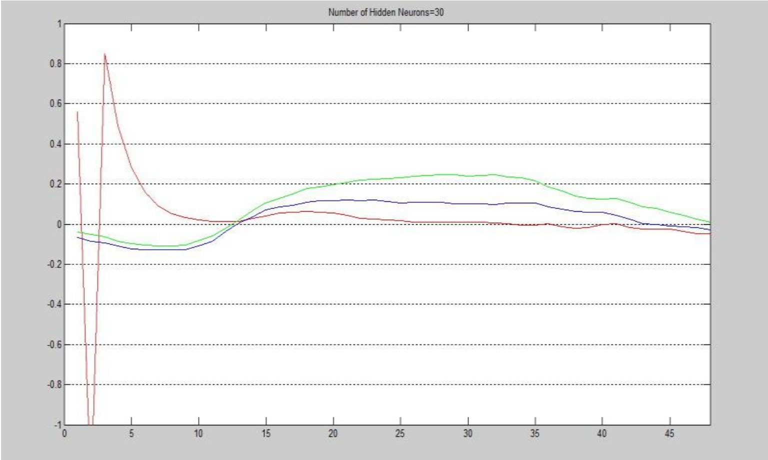


Fig 5.8 : System performance for No. of Hidden Layer Neurons (i) 10 (ii) 20 and (iii) 30.

Learning Rate Optimization:

The learning rate (alpha) is a very essential parameter that decides the rate of convergence of the network and hence the time taken for training. The performance of the neural network system was studied taking a large range of learning rates values, from 0 to 1. In case of the learning rate of 0.1, the error response was obtained to be oscillatory with learning falling behind error. For learning rate of 0.3, it was observed that the error overshoot is quite high, thus time taken to settle increases. Thus, the optimal value of learning rate is determined to be 0.2 and is kept fixed at that value for the rest of the network implementation.

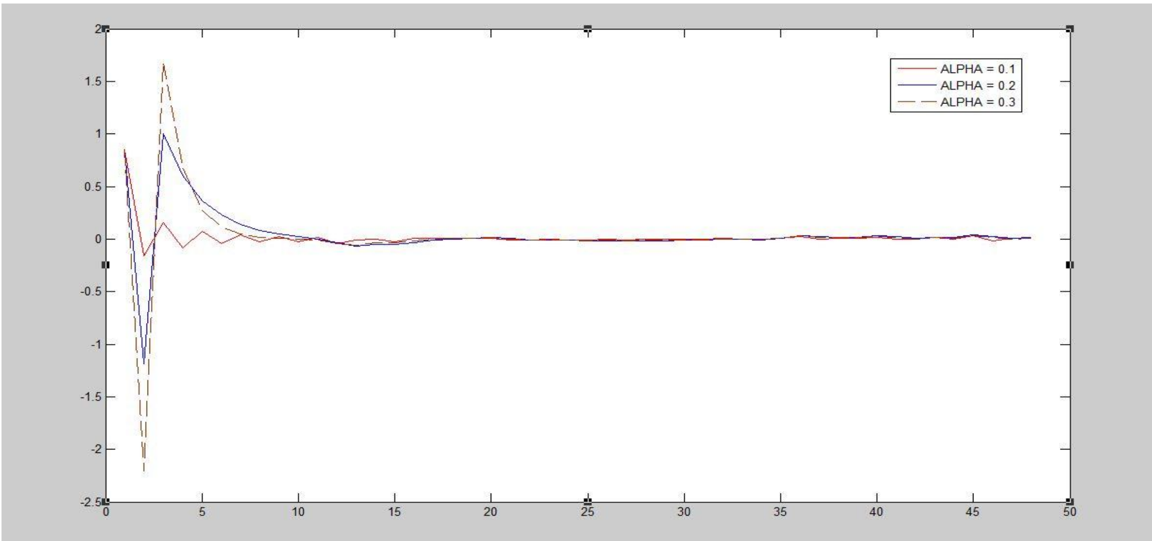


Fig 5.9: Mean Square Error plot for alpha=0.1, 0.2 and 0.3

Performance of the System:

The performance of the system is observed during training and testing phase. In the training phase, gradually the error is minimized till it reaches into allowable threshold band of errors. After that the network is exposed to partially unknown and fully unknown data. Partially unknown input data set consists of 50% of input data set that has been used for training purpose and the remainder is the data yet unseen by the network. In case of fully unknown data, entire input data set consists of data elements yet unseen by the network. The network performance was studied for these three data sets and following results were obtained.

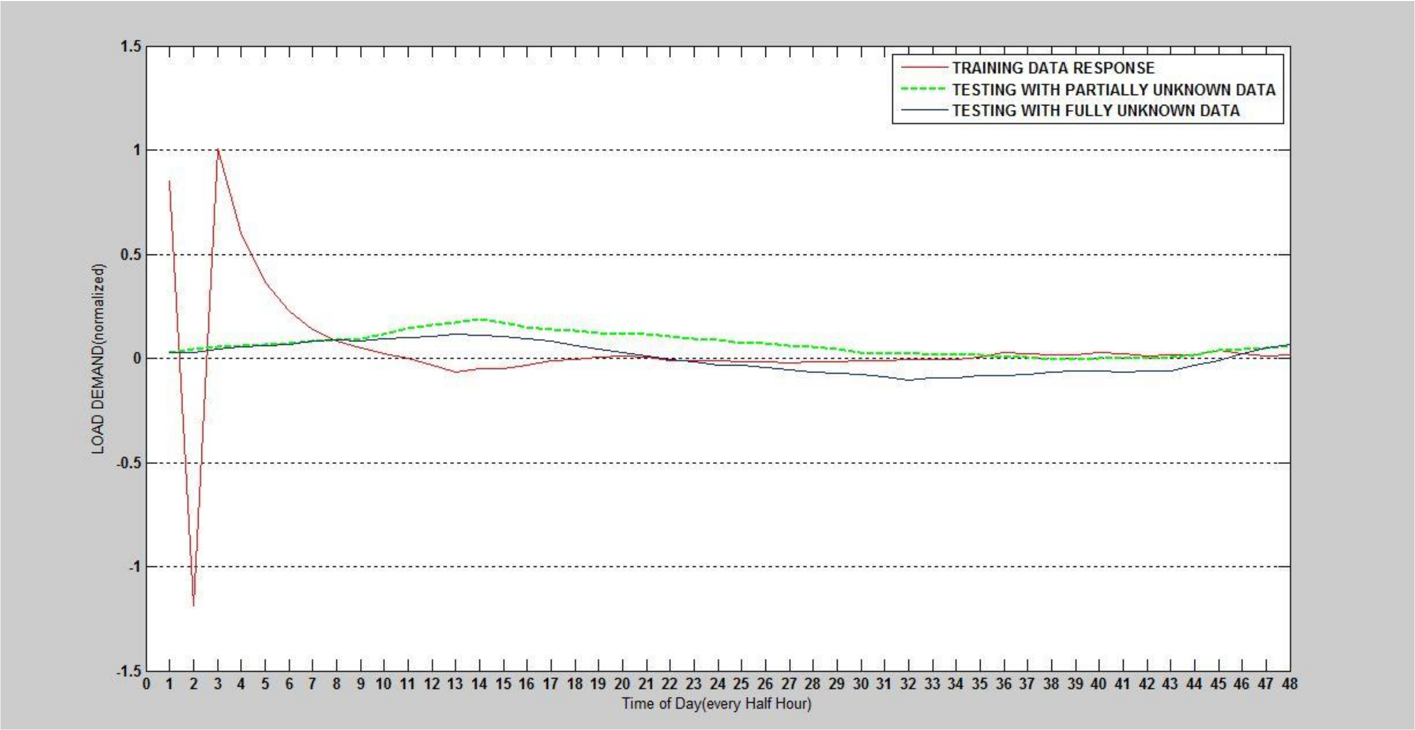


Fig 5.10: System Performance during Training and Testing stages

Chapter 6

Implementation of Load-Side Tariff Setting

6.1 Introduction

The electrical power energy produced by a power station is delivered to a large number of consumers. The tariff i.e., the rate at which electrical energy is sold is very important for electric supply company as well as consumers. **Electricity Tariff** varies widely from country to country, and also from locality to locality within a particular country. There are many reasons that account for these differences in price. The price of electric power generation depends largely on the type of the fuel used, government policies and regulation and on local weather patterns. The supply company has to ensure that the tariff should such that besides recovering the total cost of producing electricity, it also earns profit on capital investment. But the profit must be marginal.

6.2 Need for Tariff Regulation

Electricity retailers may wish to charge customers different tariffs at different times of the day to better reflect the costs of generation and transmission. Since it is typically not cost effective to store significant amounts of electricity during a period of low demand for use during a period of high demand, costs will vary significantly depending on the time of day. A determining factor for a successful implementation of a demand-based pricing model or control strategy in electricity markets is not only the effects of peak load management, but also the economical consequences for the utility operator and the end customer.

In India, the pricing has primarily been fixed/controlled by the Central and State Governments. However, this is now being done by the Regulatory Commissions at the Centre and also in the States wherever they are already functional. Power generation/ transmission is highly capital intensive and the Fixed Charge component makes up a major part of tariff. India being a predominantly agrarian economy, power demand is seasonal, weather sensitive and there exists substantial difference in demand of power during different hours of the day with variations during peak hours and off peak hours. Power demand during the rainy seasons is low in the States of Karnataka and Andhra Pradesh and high in Delhi and Punjab. Whereas many of the States face high demand during evening peak hours, cities like Mumbai face high demand during office hours. The Eastern Region has a significant surplus round the clock, and even normally power deficit states with very low agricultural loads like Delhi have surpluses at night. This situation indicates an urgent need of tariff regulation of power. This would improve utilization of existing capacities and reduce the average cost of power to power utilities and consumers.

6.3 Tariff Setting

Objectives

In order to earn profit besides recovering the cost of production of electrical energy, the tariff should include the following [23]:

- Recovery of cost of production at power station.
- Recovery of cost on the capital investment in transmission and distribution systems.
- Recovery of cost of operation and maintenance of supply of electrical energy e. g. billing etc.
- A suitable profit on investment.

Types of Tariff:

Considering the above situation, it can be seen that various methods needs to be adopted for demand based tariff calculation. Some of the existing techniques are discussed:

Demand-side management: Different means of controlling and reducing energy demand in peak periods is known as demand-side management (DSM).It includes two strategies. One strategy is **direct control** that aims to actively disconnect load. In the residential sector DSM has been achieved with direct control on water heaters, heating systems and air conditioning units within the dwellings.

Indirect load control is controlling energy demand in peak periods with strict economical incentives such as time of use (TOU) rates, time of day (TOD) rates or real time pricing (RTP) rate. The benefit of differentiated rates is that they reflect the conditions in the electricity market and to some extent transfers the risk to the electricity consumer. Comparing the above two strategies, it is concluded that the effects of differentiated pricing is relatively modest. It has also been observed that the load reduction of a large number of participating households does not coincide with the system peak. RTP projects demand highly advanced Technical solutions.

Different types of tariff are:

Traditional Tariff: It corresponds to the present method that is followed in India. It consists of two parts **Fixed part and Variable part.** Fixed part is solely related to the size of the fuse and thus this amount is kept fixed for all consumers. The flexible/variable part is an energy-based fee e.g. 2.1 cent/kWh. The energy rates are matched with mean load values with hour based resolution, kWh/h and are further aggregated into an energy tariff model.

Daily Demand Based Tariff: The second tariff is constructed as a combined solution between a flexible load-component and a fixed part based on the size of the main fuse. The fixed part of the load component tariff is to be kept at a similar level as the traditional energy tariff. The flexible part is converted to a load-based, differentiated tariff [24]. The tariff is differentiated both in a seasonal peak period as well as a daily peak period. The seasonal peak period for India can be considered as the summer months from April through October. The daily peak period is defined as the hours 7 to 19. The seasonal peak period can be charged at some rate, let say 'x'. The off peak period rate, November through March, between the hours 7 and 19, is set to half the peak rate value i.e. 'x/2'. In both cases the flexible component is matched with the monthly mean peak load value based on the three highest peaks each month.

Seasonal Demand Based Tariff: In the third implemented tariff the fixed part is completely excluded in favour of a higher flexible rate. The intention is to investigate the impact of a strict flexible construction with higher price per used kilowatt in the seasonal peak period. Again, the mean peak load values used in this case are based on the three maximum peak loads each month.

6.4 Different Tariff Calculation Techniques

1. **Simple tariff:** *when the electricity is charged at a fixed rate per unit consumed, it is called a simple tariff or uniform rate tariff.*

In this type, charges remains constant and does not vary with consumption. Energy consumed is recorded by energy meter.

2. **Flat-rate tariff:** *When electricity is charged at different uniform per unit rates for different types of consumer, it is called flat rate tariff.*

In this type different types of consumers are grouped into different classes and each class of consumers is charged at a different uniform rate. For example, the charge can be more for light loads while more for power loads. Consumers are classified on the basis of their diversity factor and load factor. Its advantage is that it is more fair for different type of consumers and also simple in calculations.

3. **<u>Two-part tariff:</u>** *When electricity is charged on the basis of maximum demand of the consumer and the units consumed, it is called two-part tariff.*

Here the total charge made to be made by the consumer is split into two components i.e. fixed charge and variable or running charge. Fixed charge depends upon maximum demand of the consumer while variable charge depends upon the units consumed by the consumer. Thus the total charge is the sum of fixed charge and variable charge.

Total charges = (b*kW + c*kWh) `

b = charge per kW of maximum demand
c = charge per kWh of energy consumed

4. **<u>Maximum demand tariff:</u>** similar to two part tariff with only difference that the maximum demand is measured by installing maximum demand meter in the consumer premises. It is mostly applied to big consumers and not applicable to small consumers.

5. **<u>Power factor tariff:</u>** *When power factor of the consumer is taken into consideration to calculate the tariff, it is known as power factor tariff.*

Power factor plays a very important role in a.c. power system. A low factor increases the rating of station equipment and line losses and hence such consumers are to be penalised.

Important types of power factor methods are:

1. **KVA maximum demand tariff**: Since KVA is inversely proportional to power factor, a consumer having low power factor has to make more contribution to fixed charges.
2. **Sliding Scale tariff**: this is an average power factor tariff in which an average power factor, for eg. 0.7 is set as reference. If the power factor of the consumer falls below this value, they are charged more, otherwise discount is given.
3. **KW or KVAR Tariff**: in this method, both active and reactive powers supplied are charged separately. A consumer having low power factor draws more reactive power and hence are charged more.

6.5 Proposed Tariff Setting based on Load

In the present project load was forecasted for the next day using neural network. Then in accordance with the load amounts predicted for the different periods of the day, tariff rates are obtained by a simple formula. First of all, the whole day is divided into five time slots, wherein the load variation was found minimum, as follows:

T1=00:00-06:00

T2=06:00-10:00

T3=10:00-14:00

T4=14:00-19:00

T5=19:00-00:00

Base load is defined as the average of forecasted load of predicted day. It is denoted by **B.L.**

Base charge is the constant charge that a consumer is supposed to pay irrespective of the amount of electricity consumed. It is charged on the reason that a consumer is continuously connected to the grid. This is charged on the basis of base load.

Current Load is the average load in the particular time zone. It is denoted by **C.L.**

Load Tariff Factor is the ratio of current load and base load.

LTF=Current Load/Base Load

The total tariff is divided into two part form.

1. **Fixed charges:** Same as Base Charge.
2. **Variable charges:** Function of Load Demand on the supply system.

   Variable Charge= (LTF-1)*C

   Here C is a constant that depends on supply conditions.

   Total Charge = Base Charge + Variable Charge

   = Base Charge + (LTF-1)*C

   If the demand is more than the base load, LTF is greater than 1 and hence TC>BC

   If demand is below the base load, LTF is less than 1 and hence TC<BC.

Approach:

As tariff is a function of load demand, following code was developed keeping Base charge as 2`/kWh and constant C as 2.5. The code was simulated in MATLAB and desired the results were obtained.

Results:

Graph 1 shows the variation of Tariff with respect to time for different hours of the day.

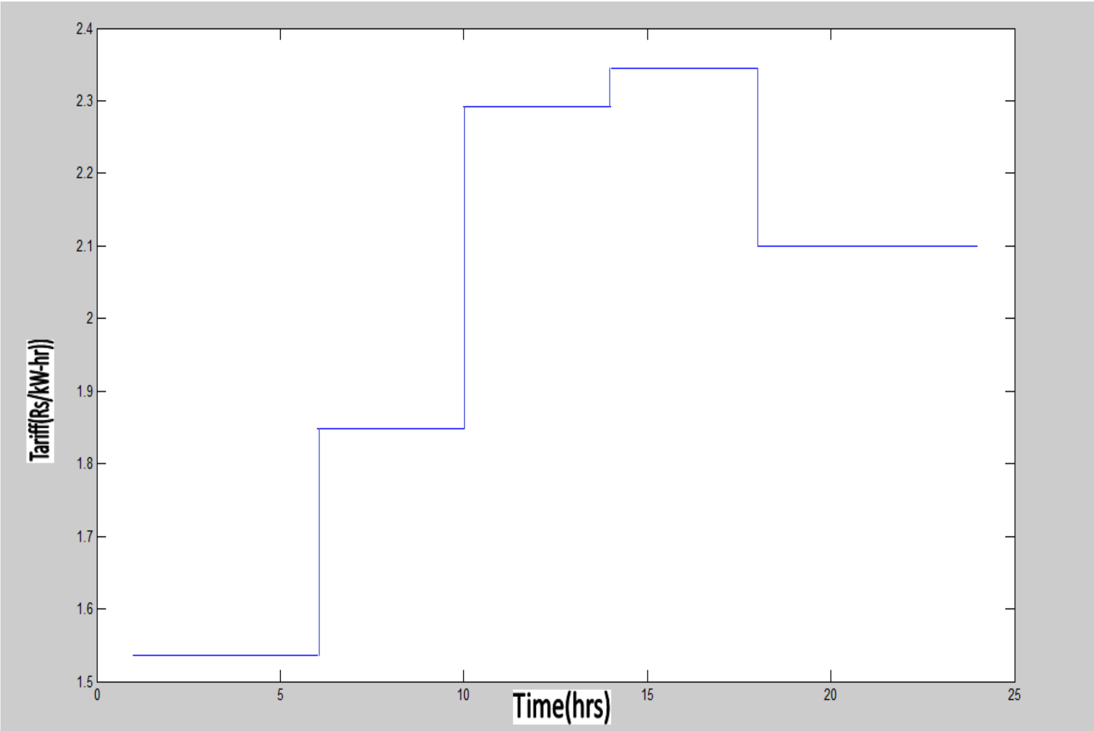


Fig 6.1: Variation of Tariff with respect to time for a Given day

Chapter 7

Development of NITR e-Power Monitoring System

7.1 Introduction to NITR e-PMS

NITR e-PMS stands for NITR e-Power Monitoring System: a one-stop for all power monitoring and analysis at NIT Rourkela. In addition to providing interested users, the load profile of the 33-KV distribution system, this portal would act as a medium of communicating tariff rates for the upcoming days. This system is based upon the data collected from the various RTUs of the Data Acquisition System (DAS). Using the raw data from these Multi Function Meters (MFMs), suitable analysis and interpretation is followed after which results are presented for display to a wider audience over the Web or the Local Intranet.

7.2 Objectives

- *To serve as a portal for dissemination of data collected by the DAS system.*
- *To enable enthusiastic users and research scholars to study the performance of the 33*
- *To serve as a mode of communication of power related information from the administration side.*

7.3 Architecture

The NITR e-PMS would comprise of an interface between the web server and the computation software, in this case MATLAB. While the web server is the key part to disseminate the data, results and analysis to an unlimited mass of users, the computation software actually calculates all the results to be shown and subsequently, interpreted. The component that serves as a communication mode between MATLAB and the web server is the Database.

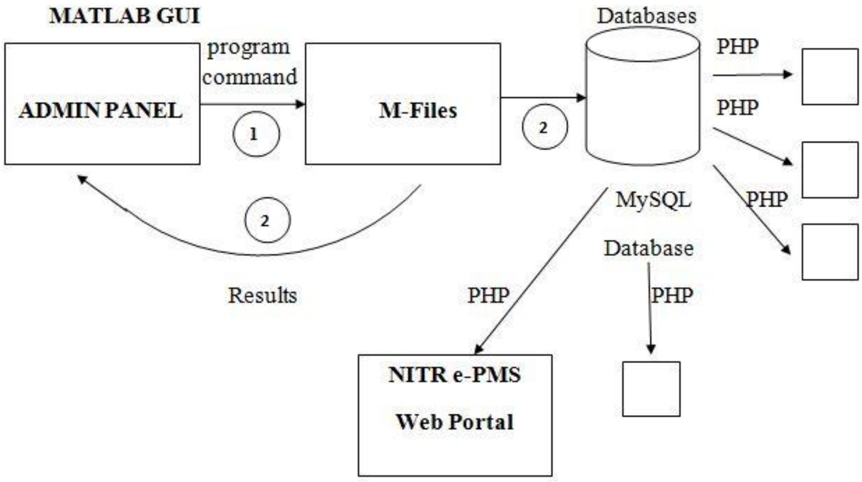


Fig 7.1: Organizational Architecture of NITR e-PMS

Thus, in essence, the NITR e-PMS is constituted of three blocks:

- MATLAB
- MySQL Database
- Web Server

MATLAB Block:

The functioning of a power management system such as the NITR e-PMS depends upon the successful processing of raw data to enable production of outputs that are desired by end users and are presented in such a manner that they are easily comprehendible. The high computational capabilities that MATLAB presents are used to process the raw data from the RTUs to predict the load and correspondingly, decide the next day tariff rates. For the purpose of initiation of the load flow calculations and the tariff settings, a MATLAB based Graphical User Interface (GUI) has been developed which enables an administrator to call the MATLAB scripts that compute the results according to the command.

As more features are added to the NITR e-PMS, corresponding changes need to be incorporated in the GUI for proper execution of all the functionalities. As of now, the Admin Panel GUI for NITR e-PMS consists of a basic GUI with two push buttons to implement the following two functions:

1. **Determine the Next Day Tariff:** Calculates the next day tariff after forecasting the load for the next day. The results are sent to the database for entry and subsequent reflection on the website.
2. **Calculate Load Flow:** This command by the user implements the load flow solution program and the load flow calculations are undertaken and subsequent display and storage of data is followed.

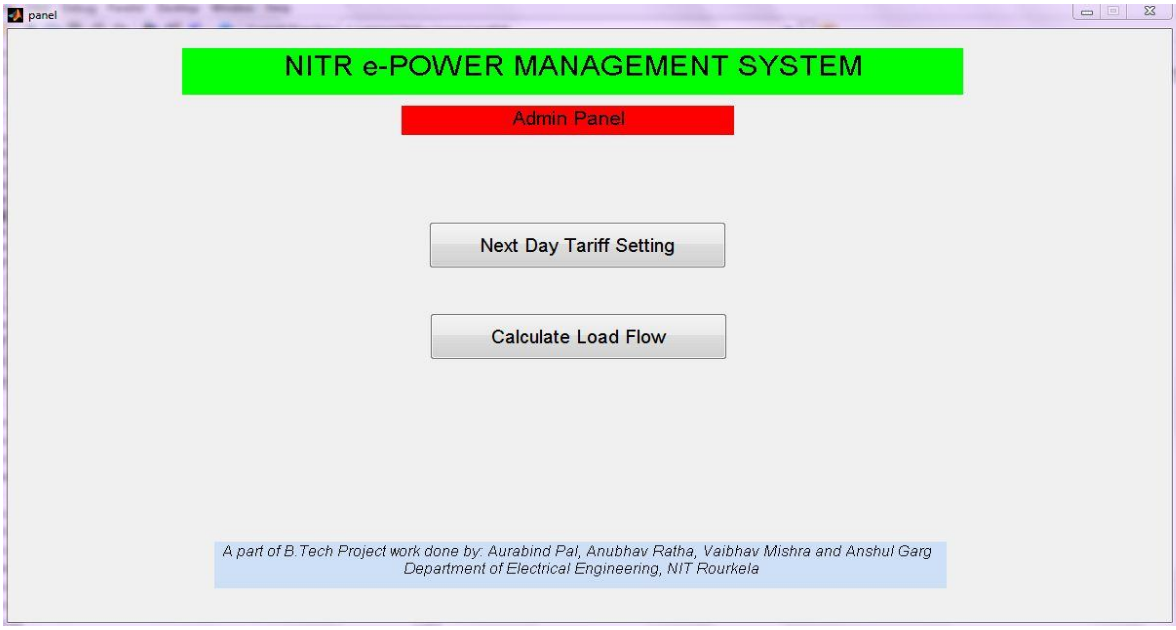


Fig. 7.2: Screenshot of NITR e-PMS Admin Panel

MySQL Database:

The database used to record the outputs of the MATLAB based calculation programs and store it for display and analysis is **MySQL Ver. 5.0.** MySQL is world's most famous open-source Relational Database Management System (RDBMS) which can be easily configured to run as a server providing multi-user access to a number of databases. MySQL can be very easily integrated with PHP which can be used to pass Structured Queries to the database and results can be published to a website.

Web Server:

The data from the database can be accessed by the web browser for multiple user display using a server side scripting language such as PHP, ASP etc. In NITR e-PMS, Hypertext Preprocessor (PHP) is the scripting language used to collect data from the database and display it in the website.

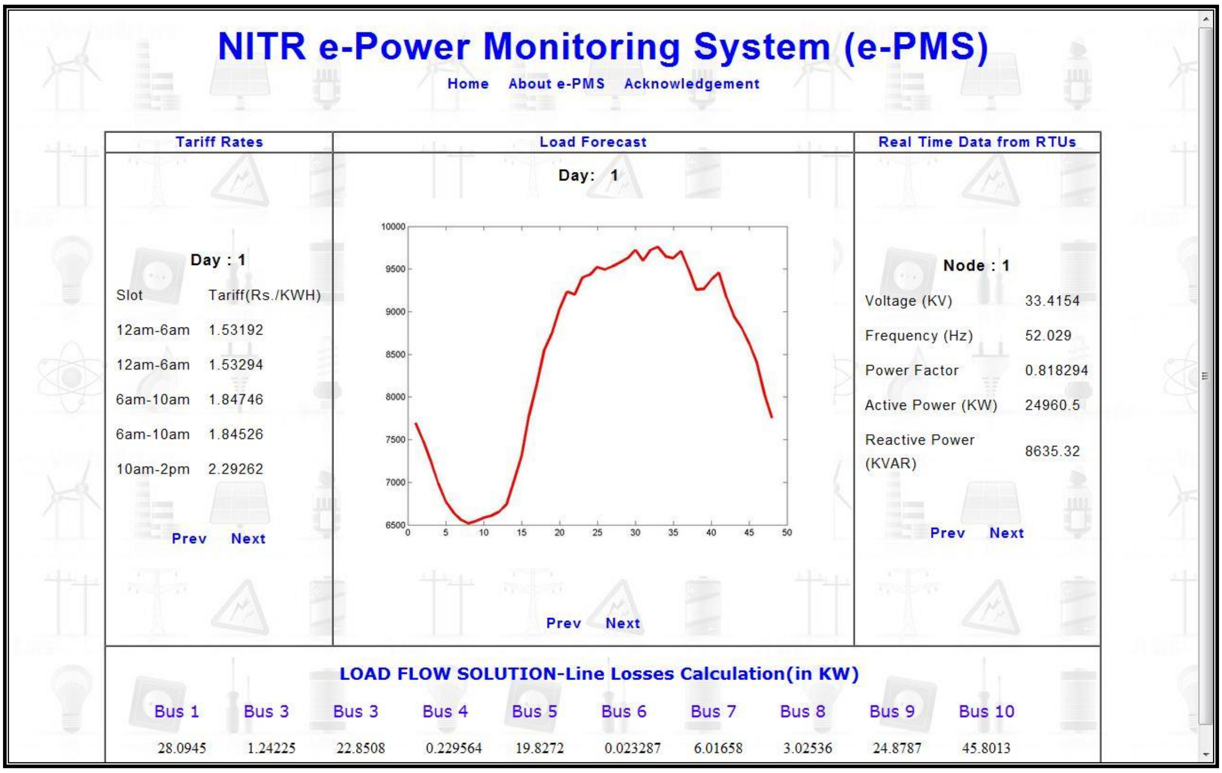


Fig. 7.3: Screenshot of the NITR e-PMS online web portal

CONCLUSION

It is expected that the present project will make the inhabitants of the National Institute of Technology, Rourkela more power conscious. It is always seen that when people see a measured change in something, they are in pursuit to change it for the better. Moreover, the project has aimed at developing a prototype model for the regional power utility companies, as it promises to make the distribution system more accountable to its losses and more automated. And we hope the present work of ours will encourage further research works in this field.

APPENDIX-I

Program for Load Forecasting using Neural Network Toolbox

```
%This Function calculates the load forecast values for the next day using
%the data of the last 30 days' load values and weather parameters.


day=2;                              %is the value of the day from which%
                                    %data is considered for forecasting%

raw_data=evalin('base','data');     %accessing data from the workspace%

    for hr=1:48                     %iterating load value for every half
                                    hour%
        p1=raw_data(hr,day:day+29); %considering load data for hour=hr for
                                    past 30 days%
        p2=raw_data(49,day:day+29); %considering THI for past 30 days%
        p=[p1;p2];
        k=day+30;
        t=raw_data(hr,k:k+29);
        [pn,meanp,stdp,tn,meant,stdt]=prestd(p,t);  %normalising input and
                                                          target data%
        [ptrans,transMat]=prepca(pn,0.04);
        net=newff(minmax(ptrans),[20,1],{'logsig','purelin'},'traingdx');
                                                %creating a feedforward
                                                 backprogaation network%
        net.trainParam.epochs=500;      %setting learning rate and epochs%
        net.trainParam.lr=0.5;
        [net,tr]=train(net,ptrans,tn);  %training the network%
        yn=sim(net,ptrans);             %simulating the network to obtain
                                                 normalised output%
        y=poststd(yn,meant,stdt);       %denormalising output%
        out(hr,day-1)=y(1,1);           %storing predicted output in out%
        tf(hr,day-1)=t(1,1);
    end

    figure;
    plot(out,'r');                      %plotting the predicted output%



%--------EXPORTING THE LOAD FORECASTING PLOT -----------%
path='C:\xampp\htdocs\epms\images\';

dow=day-1;
str1= 'day';
str2=num2str(dow);
str=strcat(str1,str2);

ext='jpg';
saveas(gcf,[path,str],'jpg');

%------------TARIFF CALCULATION BLOCK----------%
%----------Program for Tariff Setting--------------%

basel=mean(out);            %Taking the average of the day's load as
                                base load%
c1=out(1:12,1);             %Taking 12 am to 6 am as first time slot%
c2=out(13:20,1);            %Taking 6 am to 10 am as first time slot%
```

```
c3=out(21:28,1);           %Taking 10 am to 2 pm as first time slot%
c4=out(29:38,1);           %Taking 2 pm to 7 pm as first time slot%
c5=out(39:48,1);           %Taking 7 pm to 12 am as first time slot%
C(1,1)=mean(c1);           %average load of the time slot 12 am to 6 am%
C(2,1)=mean(c2);           %average load of the time slot 6 am to 10 am%
C(3,1)=mean(c3);           %average load of the time slot 10 am to 2 pm%
C(4,1)=mean(c4);           %average load of the time slot 2 pm to 7 pm%
C(5,1)=mean(c5);           %average load of the time slot 7 pm to 12 am%
k=2.5;                     %Taking the constant of load variation as 2.5%
LTF=C/basel;               %Calculating Load Tariff Factor%
tariff = 2+ (LTF-1)*k      %Setting the Tariff for the day%


 %---------------MYSQL EXPORT-----------%

conn=database('MySQL_all','root','');

for j=1:5
    slotid=j;
    coln={'Day','SlotId','Tariff'};
    vals={dow,slotid,tariff(j)};
    fastinsert(conn,'tariff',coln,vals);
end

 %-----SAVING IMAGE DETAILS IN MYSQL DATABASE-------%
s1=str;
s2='.jpg';
s=strcat(s1,s2);
col={'Day','FileName'};
val={dow,s};
fastinsert(conn,'load_fig',col,val);

clear all
```

Self Developed Program for Load Forecasting

```
function forecast = pf_try()    % Function definition line
%This Function is used for training the neural network using
%Backpropagation algorithm.
clear all
clc

%Weight and Bias Initialization %

m = 60;     % Input Layer Neurons

n = 20;     % Hidden Layer Neurons

o = 1;      %Output Layer Neurons

w1 = zeros(m,n);% Weight and Bias values initialized for Input-Hidden Layer

b1 = zeros(1,n);

w2 = zeros(n,o); % Weight and Bias values initialized for Hidden-Output
Layer

b2 = zeros(1,o);

alpha=0.2;    %Defining the Learning Rate Value
```

```
 %-----OBTAINING MAX LOAD VALUE for Data Normalization ----%

input_dataset =
xlsread('C:\Users\anubhav\Desktop\PROJECT\DATA\CONDITIONED.xls','sheet1','b
3:bi50');

loadmax = max(max(input_dataset));

 for i=1:48      %Sliding for all the 48 half hour intervals

 %-Import Data Block (Import + Normalization, Final is 'input'- the input
vector to NN-%

hr=i+2;
str1=['b',num2str(hr)];
str2=['ae',num2str(hr)];
str=[str1,':',str2];
input1 =
xlsread('C:\Users\anubhav\Desktop\PROJECT\DATA\CONDITIONED.xls','sheet1',st
r);                      %Last 30 days load value for the same TOD%
input1 = input1 ./ loadmax;         %Normalization of Input Data %


str1=['b','51'];
str2=['ae','51'];
str=[str1,':',str2];
input2 =
xlsread('C:\Users\anubhav\Desktop\PROJECT\DATA\CONDITIONED.xls','sheet1',st
r);                      % Last 30 days THI Value %
thimax = max(input2);

input2 = input2 ./ thimax;          %Normalization of Input Data %

input = [input1 input2];  %Concatenation of Load and THI Values %

%---Obtaining Desired Output Value from Excel Worksheet---%

str=['af',num2str(hr)];

do =
xlsread('C:\Users\anubhav\Desktop\PROJECT\DATA\CONDITIONED.xls','sheet1',st
r);

do = do / loadmax;

%-Import Data Block ENDS-%%- Neural Network Structure-%

      a1 = logsig(input*w1+b1);  %a1 is a 1 X n matrix
      a2 = purelin(a1*w2+b2);    %a2 is a 1 X o matrix
      e(i)=do-a2;                %Error in each iteration
      s2=-2*e(i);
      s1=(1.-a1)*(a1)'*w2'.*s2;
      dw2=-alpha*s2*a1';
      db2=-alpha*s2;
      dw1=-alpha.*input'*s1;
      db1=-alpha*s1;
      w2=w2+dw2     ;            %Updation of Bias and Weight
      b2=b2+db2;
      w1=w1+dw1;
      b1=b1+db1;
      e(i)=do-a2;
      end
      figure;
      plot(e,'r');
      hold on;
```

```
 %-----TRAINING OVER------%

  %_____TESTING______%

for i=1:48

%--Import Data Block(Import + Normalization, Final is 'input'- the input
vector to NN-- %

    hr=i+2;
    str1=['c',num2str(hr)];
    str2=['af',num2str(hr)];
    str=[str1,':',str2];
input1 =
xlsread('C:\Users\anubhav\Desktop\PROJECT\DATA\CONDITIONED.xls','sheet1',st
r);                %Last 30 days load value for the same TOD%
input1 = input1 ./ loadmax;          %Normalization of Input Data %

    str1=['c','51'];
    str2=['af','51'];
    str=[str1,':',str2];
input2 =
xlsread('C:\Users\anubhav\Desktop\PROJECT\DATA\CONDITIONED.xls','sheet1',st
r);                                 % Last 30 days THI Value %
    thimax = max(input2);
    input2 = input2 ./ thimax;           %Normalization of Input Data %
    input = [input1 input2];    %Concatenation of Load and THI Values %

%Obtaining Desired Output Value from Excel Worksheet%

    str=['ag',num2str(hr)];

do =
xlsread('C:\Users\anubhav\Desktop\PROJECT\DATA\CONDITIONED.xls','sheet1',st
r);

    do = do / loadmax;

%-Import Data Block ENDS-%

%- Neural Network Structure-%

             a1 = logsig(input*w1+b1);  %a1 is a 1 X n matrix
             a2 = purelin(a1*w2+b2);    %a2 is a 1 X o matrix
             e(i)=do-a2;

end
plot(e,'g');
hold on;

for i=1:48
%------------------------Import Data Block(Import + Normalization, Final is
'input'- the input vector to NN-------------------- %

    hr=i+2;
    str1=['ae',num2str(hr)];
    str2=['bh',num2str(hr)];
    str=[str1,':',str2];
    input1 =
xlsread('C:\Users\anubhav\Desktop\PROJECT\DATA\CONDITIONED.xls','sheet1',st
r);
    input1 = input1 ./ loadmax;         %Normalization of Input Data %
    str1=['ae','51'];
    str2=['bh','51'];
    str=[str1,':',str2];
```

```
    input2 =
xlsread('C:\Users\anubhav\Desktop\PROJECT\DATA\CONDITIONED.xls','sheet1',st
r);                                       % Last 30 days THI Value %
    thimax = max(input2);
    input2 = input2 ./ thimax;          %Normalization of Input Data%
    input = [input1 input2] ;           %Concatenation of Load and THI
Values
%Obtaining Desired Output Value from Excel Worksheet%

str=['bi',num2str(hr)];
do =
xlsread('C:\Users\anubhav\Desktop\PROJECT\DATA\CONDITIONED.xls','sheet1',st
r);
    do = do ./ loadmax;

%-Import Data Block ENDS-%

%- Neural Network Structure-%

      a1 = logsig(input*w1+b1);  %a1 is a 1 X n matrix
      a2 = purelin(a1*w2+b2);    %a2 is a 1 X o matrix
      e(i)=do-a2;
      output(i)=a2;
      target(i)=do;
end

plot(e,'b');

output = output.* loadmax;
target = target.* loadmax;
figure;
plot(output,'r')
hold on
plot(target,'g')
```

REFERENCES

[1] M.H. Haques, Efficient load flow method for distribution systems with radial and mesh configuration, *IEEE Proc-Gener Transmission Distribution.* Vol. 143, No.1, January 1996

[2] DOC.NO. - NITRKL-33KVRM-TD-01, Section-VI: Technical specification and scope of work, Part-1 Electrical Works, 33KV ring main system, SATCON consultancy.

[3] Tender Document for 33 KV Ring Main including 33/0.433 KV S/S in NIT Rourkela Campus. Part-I and Part-II, DATA Acquisition System (DAS), Doc No. NITRKL-33KVRM-TD-01, AECC Ltd.

[4] Gopal M, Digital Control and State Variable Methods, New Delhi, Tata McGraw Hills, 2008.

[5] D. Mange and M. Tomassini (Eds.), Bio-Inspired Computing Machines: Toward Novel Computational Architectures, PPUR Press, 1998, pp. 289-316

[6] Simon Haykin, Neural Networks: A comprehensive Foundation, Pearson Education Inc., 1999.

[7] Mishra Sanjib, Patra Sarat Kumar. *"Short Term Load Forecasting using computational intelligent methods"*, DSpace NITRKL.

[8] Mishra, Sanjib; Patra, Sarat Kumar. 'Short Term Load Forecasting using Neural Network trained with Genetic Algorithm &Particle Swarm Optimization, Emerging Trends in Engineering and Technology', 2008. *ICETET '08, First International Conference on;* 16-18 July 2008 Page(s):606-611.

[9] G. Gross, F. D. Galiana, 'Short-term load forecasting', *Proceedings of the IEEE*, 1987,75 (12), 1558 – 1571.

[10] A.D. Papalexopoulos, T.C. Hesterberg, 'A Regression Based Approach to Short Term Load Forecasting', *IEEE Transactions on Power Systems,* 1990, 5(1), 40 – 45.

[11] N. Amjady, 'Short-term hourly load forecasting using time-series modeling with peak load estimation capability', *IEEE Transactions on Power Systems,* 2001, 16(3), 498 – 505.

[12] W. Christianse, 'Short Term Load Forecasting Using General Exponential Smoothing', *IEEE Transactions on PAS,* 1971, 900 – 910.

[13] S.A. Villalba, C.A. Bel, 'Hybrid demand model for load estimation and short-term load forecasting in distribution electrical systems', *IEEE Transactions on Power Delivery,* 2000, 15(2), 764 – 769.

[14] K.J. Hwan, G.W. Kim, 'A short-term load forecasting expert system', *Proceedings of The Fifth Russian-Korean International Symposium on Science and Technology*, 2001, 112 – 116.

[15] A.A. Desouky, M.M. Elkateb, ‘Hybrid adaptive techniques for electric-load forecast using ANN and ARIMA’, *IEEE Proceedings of Generation, Transmission and Distribution,* 2000,
147(4), 213 - 217.

[16] K.H. Kim, H.A.Youn, Y.C. Kang, ‘Short-term load forecasting for special days in anomalous load conditions using neural networks and fuzzy inference method’, *IEEE Transactions on Power Systems,* 2000, 15(2), 559 – 565.

[17] E.A.Feinberg, J.T. Hajagos, and D. Genethliou. Load Pocket Modeling. *Proceedings of the 2nd IASTED International Conference: Power and Energy Systems,* 50–54, Crete, 2002.

[18] E.A.Feinberg, J.T. Hajagos, and D. Genethliou. Statistical Load Modelling. *Proceedings of the 7th IASTED International Multi- Conference: Power and Energy Systems,* 88–91, Palm Springs, CA, 2003.

[19] J.Y.Fan and J.D. McDonald. A Real-Time Implementation of Short-Term Load Forecasting for Distribution Power Systems. *IEEE Transactions on Power Systems,* 9:988–994, 1994.

[20] M.Y.Cho, J.C. Hwang, and C.S. Chen. Customer Short-Term Load Forecasting by using ARIMA Transfer Function Model. *Proceedings of the International Conference on Energy Management and Power Delivery,* 1:317–322, 1995.

[21] V.N. Vapnik. The Nature of Statistical Learning Theory. NewYork, Springer Verlag, 1995.

[22] M. Peng, N.F. Hubele, G.G. Karady, ‘Advancement in the application of neural networks for short-term load forecasting’, *IEEE Transactions on Power Systems*, 1992, 7, 250 – 257.

[23] Mehta V.K., Mehta Rohit, Principles of Power System.New Delhi, S. Chand and Company Ltd.,2008.

[24] F. Wallin, C. Bartusch, E. Thorin, T. Bäckstörm, E. Dahlquist. Use of Automatic Meter Readings for a Demand-Based Tariff, *IEEE, IEEE/PES Transmission and Distribution Conference & Exhibition: Asia and Pacific Dalian, China, 2005*

[25] Feinberg A. Eugene, Genethliou Dora, *Chapter 12 Load Forecasting*, State University of New York, Stony Brook